FEATURE EXTRACTION AND MATCHING OF PALMPRINTS USING LEVEL I DETAIL

PETER KITCHING BSc. MSc.

A thesis submitted in partial fulfilment of the
requirements of the University of Wolverhampton
for the degree of Doctor of Philosophy

October 2016

This work or any part thereof has not previously been presented in any form to the University or to any other body whether for the purposes of assessment, publication or for any other purpose (unless otherwise indicated). Save for any express acknowledgements, references and/or bibliographies cited in the work, I confirm that the intellectual content of the work is the result of my own efforts and of no other person.

Signature.....

Date.....

Abstract

Current *Automatic Palmprint Identification Systems* (APIS) closely follow the matching philosophy of *Automatic Fingerprint Identification Systems* (AFIS), in that they exclusively use a small subset of Level II palmar detail, when matching a latent to an exemplar palm print. However, due the increased size and the significantly more complex structure of the palm, it has long been recognised that there is much detail that remains underutilised. Forensic examiners routinely use this additional information when manually matching latents.

The thesis develops novel automatic feature extraction and matching methods which exploit the underutilised Level I detail contained in the friction ridge flow. When applied to a data base of exemplars, the approach creates a ranked list of matches. It is shown that the matching success rate varied with latent size. For latents of diameter 38mm, 91.1% were ranked first and 95.6% of the matches were contained within the ranked top 10.

The thesis presents improved orientation field extraction methods which are optimised for friction ridge flow and novel enhancement techniques, based upon the novel use of *local circular statistics* on palmar orientation fields. In combination, these techniques are shown to provide a more accurate orientation estimate than previous work.

The novel feature extraction stages exploit the level sets of higher order local circular statistics, which naturally segment the palm into *homogeneous regions* representing Level I detail. These homogeneous regions, characterised by their spatial and circular features, are used to form a novel compact *tree-like* hierarchical representation of the Level I detail. Matching between the latent and an exemplar is performed between their respective tree-like hierarchical structures.

The methods developed within the thesis are complementary to current APIS techniques.

Acknowledgements

I greatly acknowledge my Director of Studies Prof. Ian P. W. Sillitoe for his continued patience and expert guidance. I would like to thank Daren Riley, *Senior Palmprint expert, West Midlands Police*, Jason Everett, *Detective Inspector West Mercia Police* and Dianne Toyne, *Identification Bureau Manager East Midlands Special Operations Unit* for sharing their vast knowledge and insight into the forensic process which is used in the UK Police force. Special thanks are also due to Dr Raul Sutton *Associate Dean*, Dr Kevan Buckley and Prof Ian Trueman *University of Wolverhampton*. Special acknowledgements to Rita Sampainho, Cathal Kerr and the students volunteers for their work in the creation of the exemplar database. I would also like to express my gratitude to my family for their support and patience during this project. I am also indebted to my sweetheart Alice, for her support and general awesomeness.

Contents

List of Figures

List of Algorithms

List of Tables

Chapter 1

Introduction

In every day life, each human being leaves behind traces of evidence of their activities. When discovered, and properly examined, these traces can lead to proof of a person's past location. A subset of such traces are impressions left on objects by the physical interaction of the palm and fingers. Such impressions consist of perspiration and oils, due to the excretion of perspiration through pores on the palm and fingers. A variety of forensic techniques exist to recover these impressions from different types of objects. These recovered impressions are known as *latent prints*. Approximately 30% of crime scenes contain some form of latent palmprint [1].

A latent print which contains sufficient discriminating physical characteristics can be matched to an individual, hence proving that a person held or touched an object. To match a latent print, a print of the original palm or fingers is required. These original prints are known as *exemplar prints* and are stored in a database. The process of matching a latent print to an exemplar print (known as *latent to full* matching) involves individually comparing the latent print with each of the prints stored in the database, until a match is obtained. Forensic investigators have been manually applying this process to latent and exemplar prints, in order to prove that a person held or touched an object.

The use of AFIS (*Automatic Fingerprint Identification Systems*) has aided forensic investigators for more than thirty five years [2]. AFIS are used to automatically find a set of exemplar prints which best match the latent print, resulting in a ranked list of exemplar prints. There is a formal procedure, ACE-V [3] (defined by the initials of, *Analysis, Comparison, Evaluation* and *Verification*), which is followed by forensic examiners when AFIS/APIS (*Automatic Palmprint Identification Systems*) are used as an aid to guide the forensic examiner. The final authority of the match is given to the forensic examiner and must be verified by an additional and independent forensic examiner. AFIS have become a critical component of a forensic examiners toolbox. Hence, research into improving their efficacy and expanding their application is ongoing [4].

However, in practice, owing to the many means of contact between the palm and and object and the particular method of acquisition, the quality of a individual latent varies dramatically. Therefore, the complexity of the identification of the features and the matching process is not a trivial task. Hence, the use of APIS is closely supervised by forensic experts, to ensure correction feature extraction [5]. For example, when the latent print is of particularly poor quality, it is common practice amongst forensic examiners to manually mark the necessary features, prior to automatic matching and then again to verify the results [5].

Due to the similarities between palms and fingerprints, current APIS have been developed using a similar methodology as AFIS. Although it is true that there are many similarities between the features of fingers and palms, there are however, fundamental differences. It has been estimated that there is as much as 10 times more detail in palmprints than in fingerprints [2]. Due to these differences, the application of AFIS technology to palmprints does not always yield accurate matches [6].

Forensic investigators have been meticulously extending and applying their knowledge of friction skin for identification for over one hundred years. Current APIS exploit

very few of the characteristics of friction skin when performing a match. The aim of this thesis is to develop new methods for the automatic *latent to full* matching using a greater number of the friction skin characteristics (i.e. Level I detail), which exploit similar techniques to those used manually by forensic investigators.

1.1. The Use of APIS

At a visit to one of the major UK police forces forensic departments, which specialised in matching palm and finger prints from crime scenes, it was confirmed that the current APIS system, IDENT 1, requires that the features be manually marked on the latent prints prior to submission (also confirmed by [7]). It is the task of a team of forensic experts to manually identify and mark these features on the latent to be matched against exemplar database.It was also confirmed that the UK police force uses the ACE-V verification system.

It should noted that IDENT 1, in common with other commercial and academic APIS only exploit one of a number of the possible palm features (namely minutiae) in the matching process. Currently no single APIS exist which exploits *all* of the possible features typically used by a forensic examiner when attempting to identify a latent. In fact, even the best techniques described in the literature, do not make use of *wide* variety of the features within a palm used by a forensic examiner.

The need for standardisation of the entire finger and palm print matching procedure is growing [8]. International cooperation between different law enforcement agencies is expanding and has never been as important as it is at present. Hence, standardising forensic practises will ease the integration of various law enforcement departments worldwide. This includes standardising databases, procedures and the method in which any evidence is presented.

A method known as GYRO [9] (defined by the initials, *Green, Yellow* and *Red*) has been developed for presenting latent print analysis. It uses a traffic light based system based upon the forensic examiner's confidence that a feature exists; green, yellow and red for high, medium and low confidence, respectively. Although this system appears as though it has been accepted by forensic investigators [4], it has not yet been adopted by researchers in the field. No technique has adopted this methodology.

The ultimate goal for APIS is that it should be what is known as a *lights out* system. A lights out system is a fully automatic system which does not require human intervention and therefore, automatically identifies the necessary features within a latent and performs the matching. Such systems must be integrated with wider verification philosophy embodied by ACE-V

1.2. Process of Forensic Verification, ACE-V

A standard general procedure for manually associating a latent print with an exemplar print is outlined by [3]. The procedure is known by the acronym ACE-V and is comprised of the following four steps: Analysis, Comparison, Evaluation and Verification. The ACE-V procedure is adopted by forensics examiners worldwide and its purpose is two fold; to minimise any false matches between a latent and an exemplar print and to produce a standardised report which documents the entire matching procedure for its use in criminal prosecutions [4].

A *brief* review of the ACE-V procedure is now given, to show the context in which

APIS/AFIS are used in forensic applications.

Analysis The latent print(s) are analysed and any features contained within the latent print(s) are identified and marked. Analysis includes:

- the acquisition method used to obtain the latent print(s)
- the type of surface(s)
- where the latent print(s) were deposited
- the resolution of the captured image containing the latent print
- the dates and times of the acquisition and the analysis stages

Finally selected features contained within the latent prints are clearly marked and explained.

Comparison Comparisons are made between the latent prints and any exemplar prints that are deemed to match. This stage may include the results of the AFIS/APIS procedure.

Evaluation The final conclusion of the comparison stage includes details of the features of any matches. This will include side by side comparisons of any latent and exemplar prints with their features clearly marked.

Verification This step is where the analysis of an independent forensic investigator is presented, verifying the results of the original evaluation.

The techniques developed with this thesis are intended to contribute to the comparison stage, prior to verification by a forensic examiner.

1.3. Introduction to the Project

The overall aim of this project is to assess the feasibility of matching latent to full palmprints using sets of features, other than those currently used by APIS. Chapter 2 reviews the range of anatomical features present within the palm and summaries the taxonomy of the palmar features. Chapter 3 describes current APIS and shows that, APIS utilise only one of the many possible palmar features. The remaining chapters then develop novel methods for identification of alternative palmar features (chapters 5 to 7) and corresponding matching techniques (chapter 8).

The efficacy of these techniques are then illustrated by matching latent to full using an anonymised database of palm prints. The anonymised database consists of 226 left and right exemplar palmprints was created by taking inked palmprints. The database was collected from volunteer student participants at the University of Wolverhampton by Rita Sampainho during the period 25/10/12-25/03/13, under the supervision of Dr R. Sutton (Head of the Forensic Department of the University of Wolverhampton). The prints were made on standard ten print forms, kindly supplied by the West Midlands Police using permanent India ink which had been rolled onto the palm. The inked exemplars were assigned a number so that the exemplar could not be linked to the owner, then scanned by Cathal Kerr during March 2014 at a resolution of 1200 ppi

(pixels per inch) and saved using a loss less format before being destroyed to further preserve anonymity. The database used in this project was downsampled to a resolution of 500 ppi, in keeping with police standards [5].

The detailed objectives are described in the following section.

1.4. Structure of the Thesis

Chapter 2: Palmar Detail, provides the necessary background and definition of terms for all anatomical level I, II and III structures within a palm. The chapter describes the individual characteristics of the structures and their relative positions within the palm. This information is exploited in chapter 7 to automatically identify individual Level I structures and then once again in chapter 7 to produced a tree-based description of the relative positions of Level 1 structures and features groups within the palm. These are then used later in the matching procedure.

Chapter 3: APIS, provides a review of the use of level I, II and III detail within APIS. It is shown that the dominant form of detail used within APIS is Level II (i.e. minutiae). It is further shown that Level I structures are only used within AFIS, where they are regarded as singularities within the orientation field and are not used in the matching procedure. The chapter then goes on to introduce the general stages of APIS processing, which are derived from previous AFIS (i.e *Enhancement, Feature extraction* and *Matching*). These stages are then used within the chapter to categorise previous work. The same stages are then used within thesis structure to categorise our methods. A brief description of minutiae processing is provided to allow for later comparison of orientation field extraction methods. Two methods are chosen for comparison with the work in the thesis.

Chapter 4: Circular Statistics, provides a brief description of circular statistics which is a necessary basis for the formal framework used in chapter 5 and 7, when processing the orientation field and extracting features. The chapter shows that the use of circular statistics provides a more compact and efficient characterisation of local variation in the orientation field, than the ad-hoc method used in AFIS [6].

Chapter 5: Orientation Estimation, Orientation field estimation forms the basis of all AFIS and APIS processing. Chapter 4 introduces a novel filter technique which improves the estimation accuracy of the orientation field. This technique uses the amplitude to the orientation estimate to remove unreliable estimates. Previous work has often used isomorphic kernel for the estimation of the orientation field. Chapter 4 introduces a methodology for the comparison of orientation estimation techniques. This methodology is then used to validate the derivation of an optimal frequency selective 3 by 3 isomorphic kernel for the estimation of the orientation field in palms. The performance of the new kernel, when compared with previous work, is shown to have a lower average error by a factor 10 over some wavelengths, over the friction ridge frequency range. However, the noise immunity of small isomorphic kernels is poor.

The work described in the chapter then improves upon the accuracy of the optimal isomorphic kernel by introducing spherical quadrature based filters (SQF) as orientation estimation operators. To the author's knowledge there exist no previous comparative study of the accuracy of SQF as an orientation operator. This chapter provides a normalisation procedure which allows comparison of a number of alternative SQFs and isolates the optimal SQF and the corresponding standard deviation of the filter for

friction ridges.

Chapter 6: Orientation Estimate Filtering, introduces a novel adaptive filtering and interpolation technique which is used to estimate the orientation of the previously unreliable estimates, removed during chapter 5. The orientation of the Level I detail of the friction ridges is then calculated. The performance of the overall enhancement technique (i.e. chapters 5 and 6) is then illustrated by comparison with previous techniques to show better classification and positional estimates of Level II details.

Chapter 7: Feature Extraction, introduces the first use of circular variance for segmentation , in which regions of high variance are used as a means of identifying level I structures (i.e deltas, loops and whorls). These structures are located by their centres and classified by the properties of the circular statistics of a radial sweep about their centres.

The chapter then introduces two hierarchical tree-based representations (*up and down trees*) of the same palm or latent, in which a node within either tree describes a local homogeneous region of circular variance. Therefore the nodes within the span of the trees have the same values of circular variance and each root corresponds to the variance of the valid area of the palm print. The trees differ in that one is derived from the circular variance of the palm and the other from its complement. The nodes within the respective trees represent non-intersecting regions within the palm and therefore provide complementary views of the same data set. Each node contains features representing a particular region and the contents of node are used in the matching process.

Chapter 8: Matching,This chapter describes matching techniques which are invariant to translation and rotation for the features derived in chapter 7. The chapter consists of two main sections. Firstly, individual matching procedures are developed for loops and deltas, based upon their *paths,* which are represented as lists of Cartesian coordinates. The section creates invariant features for the lists, which are then used to match the individual structures.

Secondly, the *up and down trees* for the latent and exemplar are matched on a canvas, which is iteratively filled using the best match nodes within the trees. The final matching score is determined by the correspondence between the canvas and latent.

Chapter 9: Results, describes the results of matching pseudo-latents to *any* part of the palm, using *up and down tree* matching. The results are then refined to investigate the performance of matching within the major regions of the palm. The chapter goes on to illustrate the matching of deltas and loops. Finally the combined operation of both methods is illustrated .

Chapter 10: Conclusions and Further Work.

Chapter 2

Palmar Detail

This chapter provides the necessary background and definition of terms for all anatomical Level I, II and III structures within a palm. The chapter is structured as follows. Section 2.1 provides a review of the anatomical properties of a palm. This includes the factors involved during the formation of the palm, which result in different sets of features which are unique to each individual palm. Section 2.4 shows how individual palms can be classified based upon the frequency and the relative position of a specific type of feature. Section 2.4 discusses where particular patterns of features exist within regions of the palm, particularly Level I detail which are the subject of the thesis.

This chapter concludes with a tree diagram which illustrates the taxonomy of the features of a palm.

2.1. Introduction to the Anatomical Structure of a Palm

Due to various biological and climatic factors, each individual palm exhibits unique anatomical features which can be used for personal identification [10]. This section discusses the biological factors that influence the anatomical structure of a palm and hence, the resulting features contained in exemplar or latent palmprints.

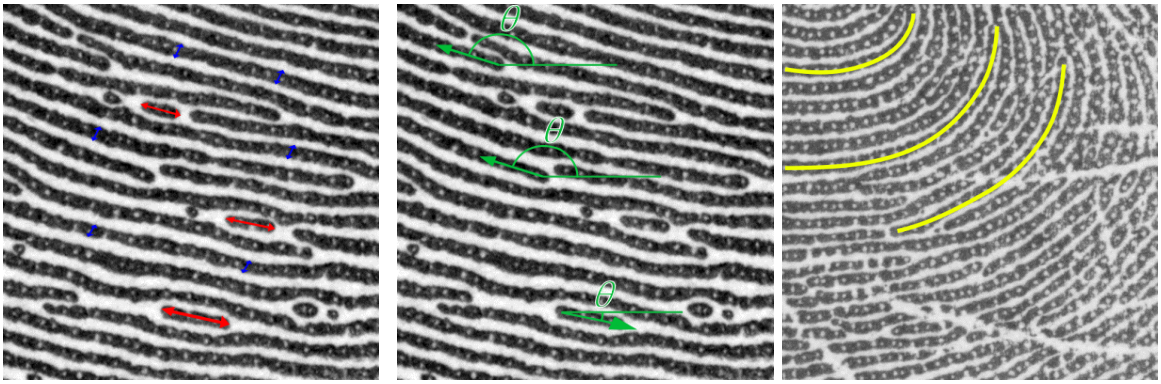
The surface of the human palm is dominated by pseudo-parallel ridge like structures. These ridge like structures are known as *friction ridges* and are illustrated in figure 2.1.

Friction ridges develop due to the buckling instability acting on the basal layer of the foetal epidermis [11]. The term *friction ridge* is adopted for these ridges because they exist on areas of the epidermis known as *friction skin*.

Friction skin is defined as being an area of skin which aids in the ability to grip and manipulate objects and can be identified by the absence of hair and the presence of sweat glands [12].

Friction ridges have individual attributes such as: length, breadth (this can also be interpreted as density), depth, direction (also known as orientation) and curvature. A example of what is meant by these terms is illustrated in figure 2.1. The red arrows indicate the length of a friction ridge, which is the length from start of the friction ridge to its end. The blue arrow illustrates the breadth or the density, this is the width of each friction ridge at any particular point along a friction ridge. The direction of the friction ridge is illustrated by the angle in green, it is the angle that the structure of the friction ridge takes compared to the image plane. The angle of the friction ridge which is also known as *ridge flow,* is measured using polar coordinates, hence horizontal friction ridges are zero radians and vertical friction ridges are $\frac{\pi}{2}$. The curvature of a friction ridge is a measure of how quickly the orientation of the friction ridge changes, high curvature is illustrated by the yellow lines.

In the physical world, whenever tension and compression occur on a semi-flexible membrane, ridges occur, aligned perpendicularly to the tension across a surface [13]. At the time of friction ridge development (when the foetus is in the sixth-seventh week of gestation), *volar pads* begin to appear on the palm [14]. Volar pads are defined as being transient swellings of mesenchymal tissue under the epidermis. Due to the stress on the epidermis caused by movement over the volar pads, groups of adjacent friction ridges are formed that share similar attributes. At certain points on the epidermis



(a) Breadth (density) - blue, length - red

(b) Orientation - green

(c) Curvature - yellow

Figure 2.1.: The attributes of friction ridges. The blue lines indicate the breadth (friction ridge frequency), the red lines indicate length, the green lines indicate the orientation and yellow indicates the curvature.

above the volar pads, different sets of friction ridges intersect, these intersecting sets of friction ridges are known as *Level I structures*. These intersecting friction ridges develop structures in a manner that can be modelled mathematically, where the size and shape of the volar pad is known [11]. Level I structures are discussed further in section 2.3.

Regions of friction ridge flow are delineated by *flexion creases*. *Flexion creases* are deep valley like structures that are created by a combination of the interaction of the movement of the epidermis and the position of the volar pads, the resulting deformations in the skin and muscle areas producing folds in the skin [8]. Flexion creases are created as soon as the unborn foetus starts clenching its hands, which is when the foetus is in the twelfth week of gestation. [13] suggests that both flexion creases and friction ridges undergo simultaneous development, due to the physical modification of ridge flow to accommodate flexion creases. Three types of flexion creases are distinguished, namely, *major flexion creases*, *secondary flexion creases* and *minor creases*. Major flexion creases are defined in section 2.1.1, secondary flexion creases and minor creases are defined in section 2.3.6.3.

Due to the complex nature of foetal epidermis development, both the size and shape of flexion creases and the pattern of friction ridges provide unique features that differ between palms [15].

Although these features are unique to particular palms, friction ridges and flexion creases tend to follow a general pattern, which can be used to segment any particular palm into major regions.

2.1.1. Segmentation of the Major Regions of the Palm by the Major Flexion Creases

There are three *major flexion creases* which are each illustrated by particular colours in figure 2.2 (a), namely:

1. *Distal Transverse flexion crease- red*
2. *Proximal Transverse flexion crease- green*

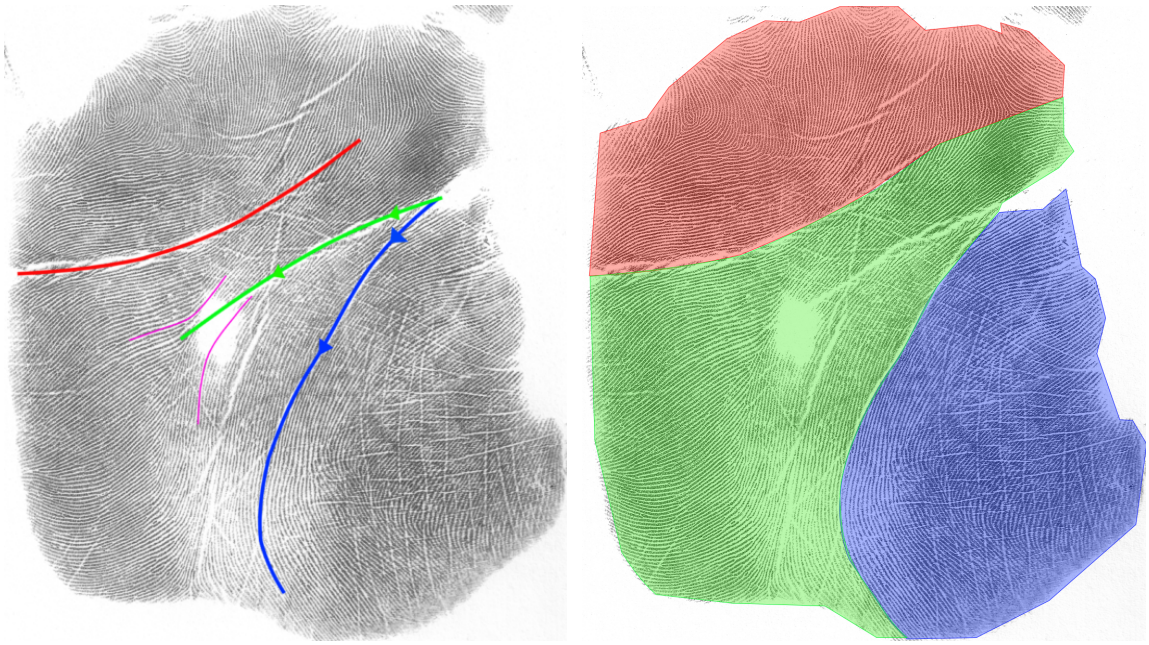
3. *Thenar Crease flexion crease- blue*

Smaller, less pronounced creases , known as *minor* and *secondary flexion* creases, also exists in a palm. These are discussed further in section 2.3.6.3.

The distal transverse and the thenar creases separate the three regions of the palm as shown in figure 2.2 (b):

1. *Tri-Radiate region- red* (also known as *Base* or *Interdigital Pads*)
2. *Hypothenar region - green*
3. *Thenar region- blue*

The colours in figure 2.2 (b) illustrate the location of the above three regions. The tri-radiate region (red) is the area beneath the base of the fingers and above the distal transverse crease. The hypothenar region (green) is the area spanned by the distal transverse, thenar crease and the edge of the palm. The thenar region (blue) is the area that spans the thenar crease and the base of the thumb.



(a) Distal transverse (red), Proximal transverse (green) and Thenar crease (blue)

(b) Tri-radiate (red), Hypothenar (green) and Thenar (blue)

Figure 2.2.: The regions and major flexion creases of the palm

Proximal Transverse Crease The proximal transverse crease enters the palm either directly above or at the same position as the thenar crease, as illustrated by the green arrows in figure 2.2 (a). The crease travels through the top quadrant of the hypothenar region, ending in what is known as the *funnel* area at the centre of the palm in the hypothenar region; the funnel area is illustrated in purple in figure 2.2 (a).

Thenar Crease The thenar crease enters the palm either directly below or at the same position as the proximal transverse crease, as illustrated by the blue arrows in figure 2.2a (a), it forms an arc with the base of the muscles of the thumb. The crease follows a similar arc to the ridge flow in the thenar region.

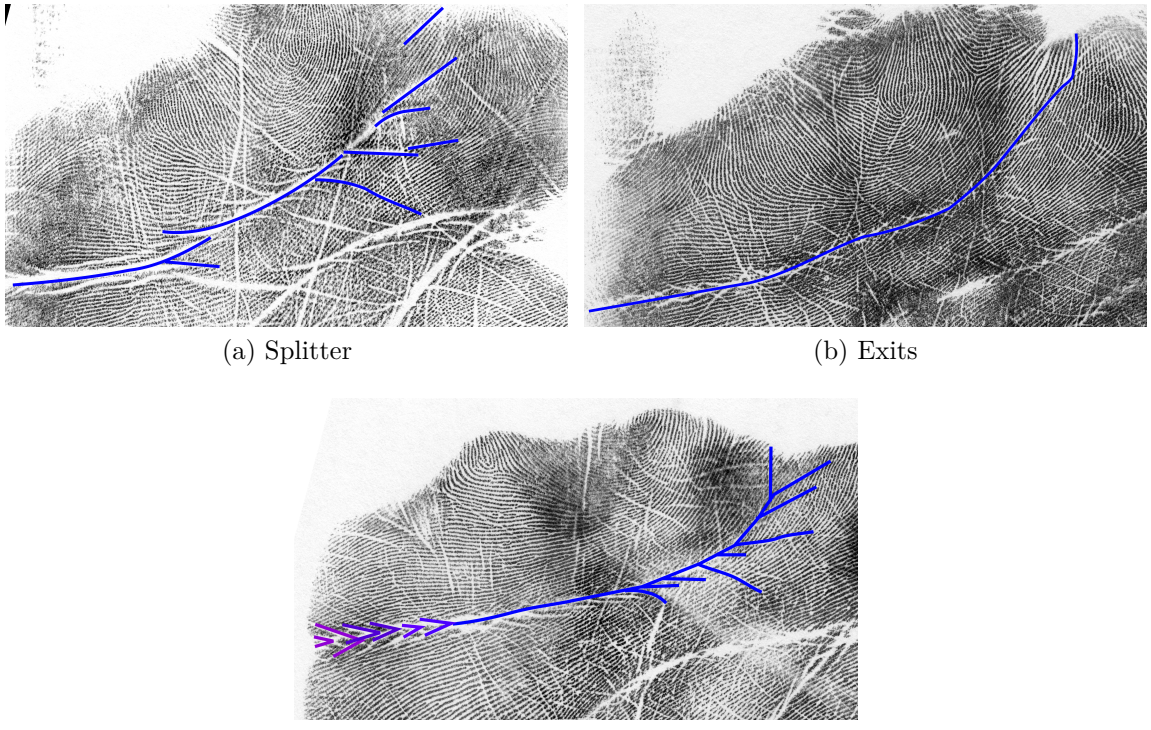
Distal Transverse Crease The distal transverse crease enters the palm below the little finger and travels into and up the palm, usually disappearing around the fore/second finger, illustrated in blue in figure 2.2 (a). It is common for many bifurcation points to exist along this crease, even to the extent that it completely splits into two or more large creases [7].

If these bifurcation points exist towards the edge of the palm, they often form a row of V's, known as *crows feet*. Crows feet open out towards the outside of the hands as illustrated as the distal transverse fade into purple in figure 2.3 (c).

Variations in the distal transverse crease are common. Beneath the fore/second finger the distal transverse crease tend to do one of the following:

1. Bifurcates into other large creases and terminates; a type known as a *splitter*. Illustrated in figure 2.3 (a).
2. Continues up the palm and exits the palm around the fore/second finger; a type known as *exits*. Illustrated in figure 2.3 (b).
3. Bifurcates, but continues to exit the palm; a type known as *bifurcates*. Illustrated in figure 2.3 (c).

These are the terms used by British police forces [7], to distinguish between different types of distal transverse crease.



(a) Splitter

(b) Exits

(c) Bifurcate

Figure 2.3.: Common variations of the distal transverse, highlighted by the blue lines.

The following section describes the detailed attributes of the friction ridges within the thenar, hypothenar and tri-radiate regions as segmented by the major flexion creases

2.2. Friction Ridges

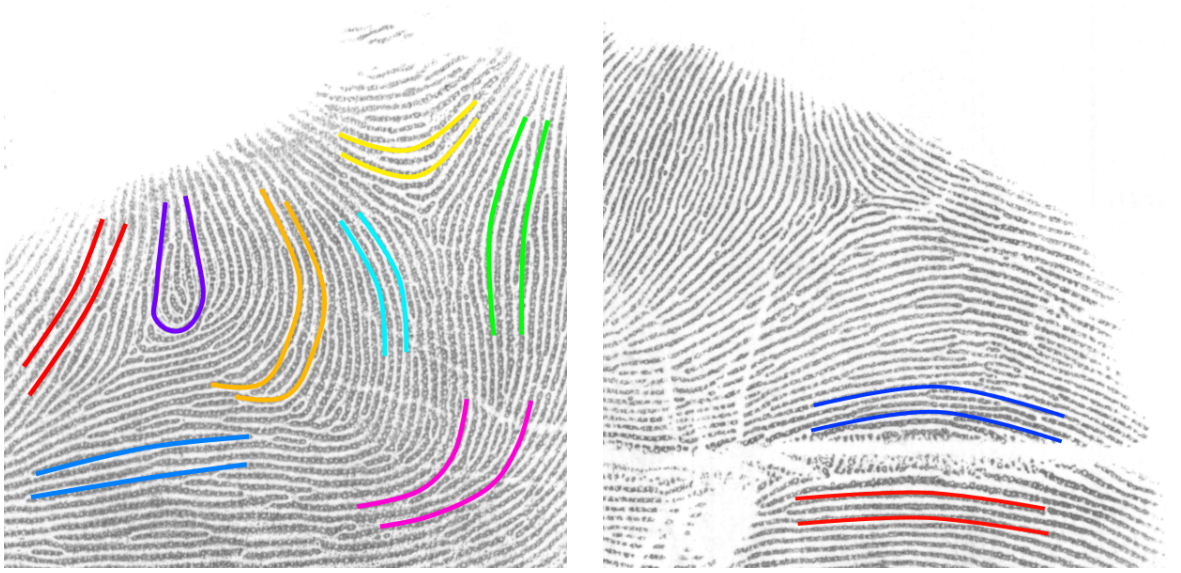
Each friction ridge is constructed from adjacent continuous *friction ridge units*. A friction ridge unit is defined as being a single pore in a single section of a friction ridge [8], friction ridge units are illustrated in figure 2.17 (c). Sequences of friction ridge units coalesce to form friction ridges. The quantity of friction ridge units that join to form a friction ridge is established at random [8] during the early stages of friction skin development. Hence, where a friction ridge starts or stops is unique to the individual palm [8]. The end points of friction ridges are a subset of what are known as minutiae points and are discussed further in section 2.3.5.

Friction ridge flow in palmprints can be used to classify the regions of the palm. As previously stated, the attributes of friction ridges (including length, frequency, orientation as described in section 2.1) are unique to each palm, however there are common traits that friction ridges exhibit in the major regions. These common traits can be used to identify which major region a segment of a palmprint has originated from. The following subsections discusses the attributes of friction ridges which are *unique* to each major region of the palm.

2.2.1. The Tri-Radiate Region

Within the tri-radiate region the ridge flow is almost entirely dominated by groups of intersecting friction ridges. An example of these different groups of friction ridge flow is illustrated in figure 2.4 (a). Additional structures, namely deltas, whorls and loops, are formed in this region owing to the volar pads. These structures are known as Level I structures and are discussed further in section 2.3. The structures in this region and the resulting ridge flow can be of high significance for matching purposes because this region of the palm is often used to grab or hold objects [16] and latent prints from this region are often found at crime scenes [1].

The ridges immediately above the distal transverse beneath the little finger, form a slight arc, producing a mound like appearance in the adjacent friction ridges above; known as *mounded*. The friction ridge flow below these mounded friction ridges, in the upper part of the hypothenar, below the distal transverse crease are flat. Hence these friction ridges known as *mounded over flat*; these are illustrated in blue and red respectively, in figure 2.4 (b).



(a) Different groups of friction ridge flow

(b) Mounded over flat; illustrated by the blue and red lines respectively.

Figure 2.4.: Common friction ridge flow in the tri-radiate region

The epidermis in this region is protected by a layer of subcutaneous fat [16] and so when the tri-radiate is put under pressure, it produces additional distortions in the deposited latent palmprint. These distortions complicate any matching procedure.

2.2.2. The Hypothenar Region

The hypothenar region contains the largest area of unimpeded ridge flow on the palm [12]. If two friction ridge paths can be traced out in the hypothenar, as illustrated in figure 2.5, they are said to form a *large* funnel. The funnel is said to be located at the point at which the ridges flow out from the centre of the palm, known as the *neck* of the large funnel. The area at the neck of the large funnel is an area of particular interest for identification, since there are many friction ridges that do not bifurcate but end at that point. The end of a friction ridge is a subset of what are known as *minutiae points* (These are discussed further in section 2.3.5since they are the basis of current APIS/AFIS techniques).

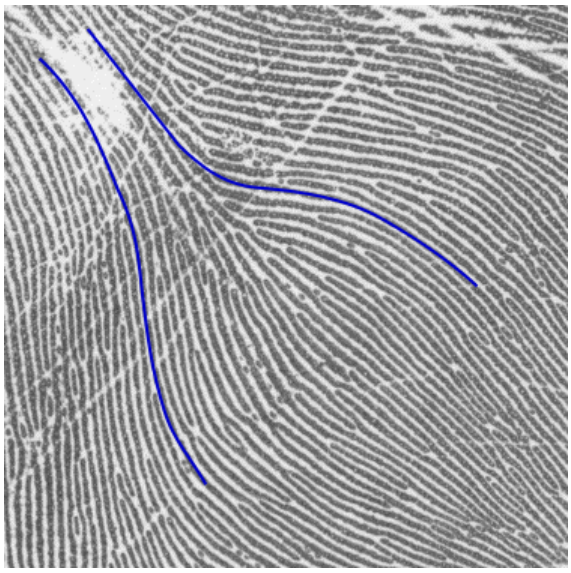
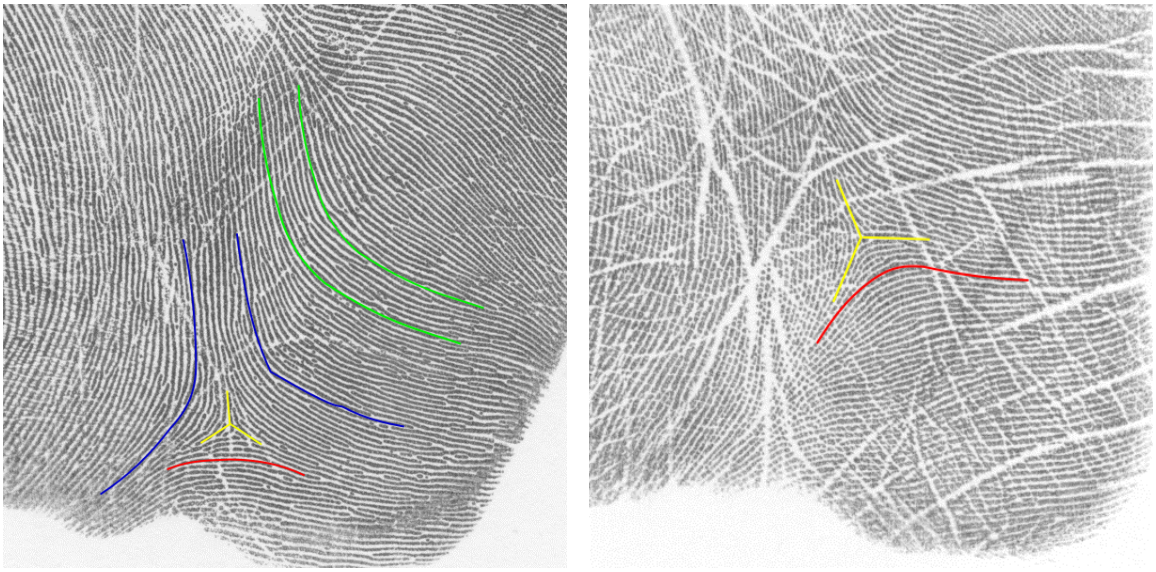


Figure 2.5.: Funnel in the centre of the hypothenar region

The *carpel delta* is a specific type of structure which usually exists close to the wrist at the base of the palm in the hypothenar region. Deltas are discussed in more detail in section 2.3. A typical position for the carpel delta is illustrated in yellow in figure 2.6 (a). When this lower carpel delta exists, the ridge flow in the hypothenar marks out a large sweep across the centre of the palm. This is known as the *belly out* [7]. The belly out is illustrated in green in figure 2.6 (a).

The area immediately adjacent to the thenar region normally has an almost parallel set of ridges that form an upside down funnel, when traced down the palm. This funnel is illustrated in blue in figure 2.6 (a). Usually the carpel delta is found in the cone area of the funnel and the flow of the ridges in this area is dependent upon the position of the carpel delta.

The ridge formation below the carpel delta is known as the *hump,* as the ridges form a *smooth hump.* A smooth hump is illustrated in red in figure 2.6 (a). In the case of a higher carpel delta, the hump is larger in size, this is illustrated in figure 2.6 (b), where the *higher carpel delta* is shown in yellow and the mounded friction ridges in red. Also, for a higher carpel delta, the ridge formation on the thenar side tends to be smooth whilst the ridge flow on the hyperthenar side tends to be sharper; this is known as the *sharp over smooth.*



(a) Low carpel delta (b) High carpel delta

Figure 2.6.: Friction ridge flow in the hypothenar, blue illustrates the upside down funnel in the region, yellow illustrates the carpel delta, red illustrates the hump beneath the carpel delta and green illustrates the belly out.

The area directly below the proximal transverse has some unusual small ridges. These ridges appear to crash into the crease; they are known as *suicide ridges* and do not exist above the crease (in the tri-radiate). Suicide ridges are illustrated in red crashing into the proximal transverse in blue in figure 2.7.

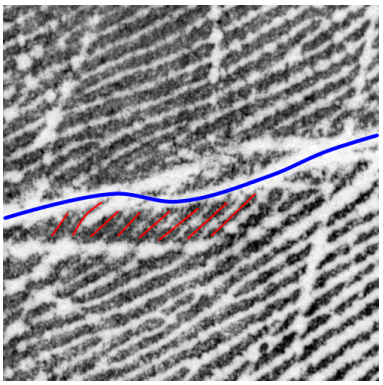
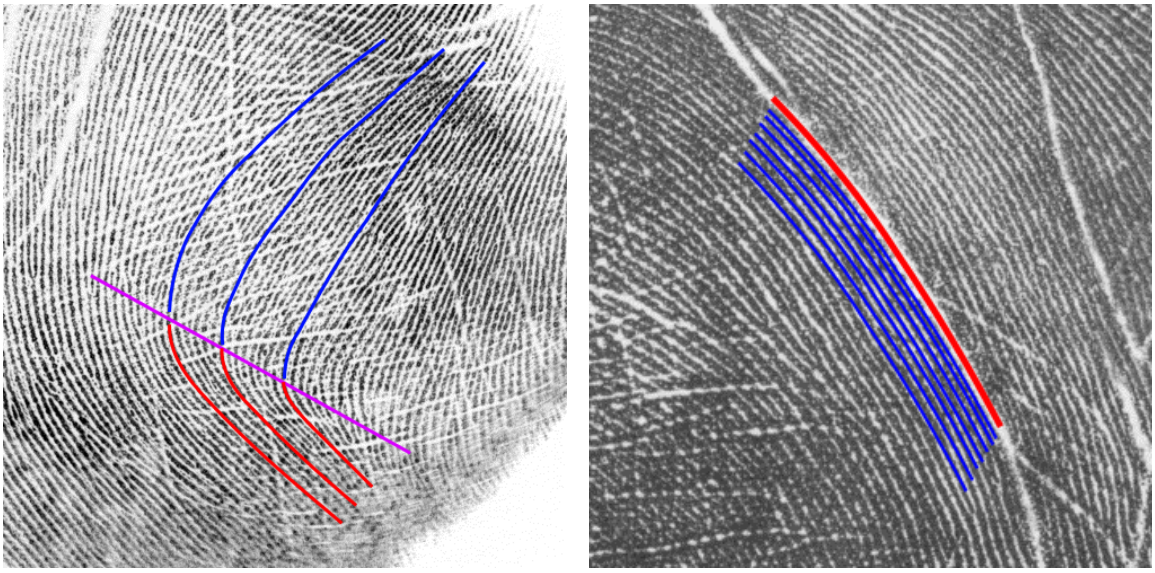


Figure 2.7.: Suicide ridges (red), below the proximal transverse (blue)

The hypothenar region is of particular interest for forensic examiners as latent palmprints from this region are the most common at crime scenes [1].

2.2.3. The Thenar Region

The friction ridge flow in the thenar region generally spans the thenar in a semi-circular arc around the base of the thumb. The flow follows the shape of the base of the thumb and the thenar crease. The point at which the arc's curvature is the greatest (illustrated in purple in figure 2.8 (a)) is of interest. If a line is drawn about the point where the arc turns the most, it splits the arc into two. This artificial split in the ridge flow separates



(a) The Long and the Short, illustrated by the blue and red lines, respectively

(b) Semi-circular arc, where the density of the friction ridges (blue) is higher as they approach the thenar crease (red)

Figure 2.8.: Ridge flow in the thenar region

what is known as *the long* and *the short* by forensic examiners [7, 17], illustrated in figure 2.8 (a), by the blue and red lines, respectively.

The very distinct ridge flow in this region of the palm, make it one of the regions used by forensic examiners for correct region identification and also orientating the latent print.

The distance between the friction ridges (which dictates the ridge density) tends to reduce the further away from the thenar crease (lower frequency), The closer to the thenar crease, the spacing between the friction ridges increases (higher frequency). This increase in friction ridge frequency is illustrated in red in figure 2.8, where the thenar crease is illustrated in red.

The following sections 2.3 to 2.3.6 illustrate what forensic examiners define as Level I, II and III detail.

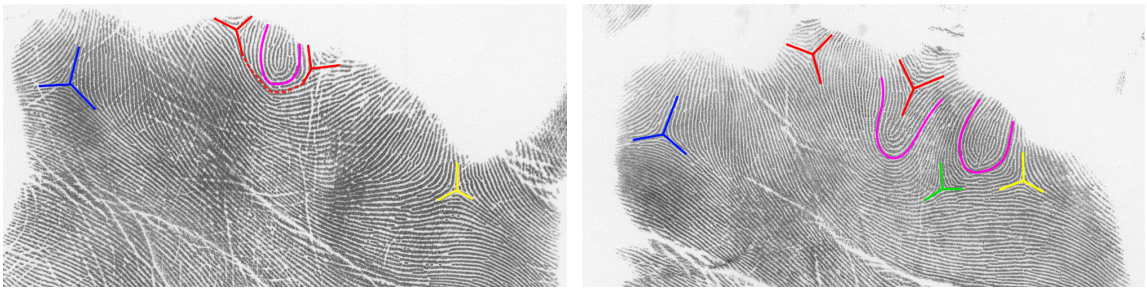
2.3. Level I Detail

The attributes of friction ridges namely, the orientation, the density and the curvature, are considered Level I detail. Particular patterns of Level I detail give rise to what are known as Level I structures. Level I structures are further classified into subgroups: *deltas*, *loops, whorls* and *vestiges*. Each type of Level I structure has its own unique pattern of intersecting friction ridges, known as friction ridge patterns [18]. The classification system for Level I structures is described below.

2.3.1. Deltas

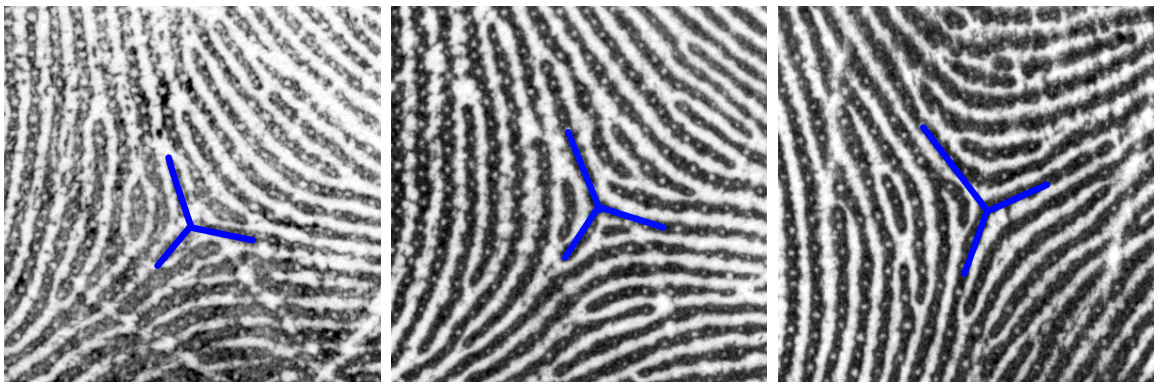
A delta, also known as *triradius*, is the meeting point of three differently orientated parallel friction ridge sets. The deltas, shown in blue in figure 2.10, have a central point from which three friction ridges emanate. In 95.2% of palms within the Canadian police database investigated in [18], there were at least four deltas in the tri-radiate, these

deltas usually exist below the base of each finger.



(a) Four Deltas (b) Five Deltas

Figure 2.9.: Structures within the tri-radiate, clean deltas (blue), cone deltas (red), upside down capital T delta (yellow), extra delta (green), connected deltas (dashed red) and loops (purple).



(a) (b) (c)

Figure 2.10.: Deltas illustrated in blue.

In general, the properties of a delta are dependent upon the position in which it is found within the palm and they can be most readily identified by the angles made by the friction ridges at the central point. The classification system for deltas used by the UK police force is illustrated in figure 2.11 and is explained below.

Clean delta The most prominent of the deltas is known as the *clean delta [7].* The clean delta exists in the triradiate region, beneath the index finger, which is denoted 1.1 in figure 2.11. It is called the clean delta for two reasons: 1) the angles made by the friction ridges that flow from the centre of the structure are approximately equal 2) the delta exists in an area of the palm where typically, there are are few flexion creases, so the friction ridges in this area tend to be readily visible.

Cone delta The *cone delta* is the most common type of delta on the palm. Typically, cone deltas exist beneath both the second and third finger, denoted as 1.2 in figure 2.11. The name comes from the appearance of the delta, since it has the shape of an ice cream cone, it is also sometimes called the cocktail glass delta for the same reason [7]. The cone delta is often use to orientate a latent palmprint, since the cone delta is

upright; when the base of the palm is at the bottom and the triradiate is at the top, as in figure 2.11. When the palm is aligned in this manner, the relationship of the angles of the friction ridges is shown in figure 2.11.

T delta Typically the *T delta* exists beneath the base of the little finger in the triradiate region. The name comes from the appearance of the delta, since it resembles an upside down capital T [7]. In a similar manner to the cone delta, a T delta can be used to orientate a latent palmprint. The T delta is denoted 1.3 in figure 2.11, where the relationship of the angles of the friction ridges is also shown in figure 2.11.

The *clean*, *cone* and *T* deltas (1.1, 1.2 and 1.3 in figure 2.11, respectively) all exist in the triradiate region of the palm. Although this section discusses the typical case, there are many different configurations of these deltas in the triradiate. These are discussed further in section 2.4.

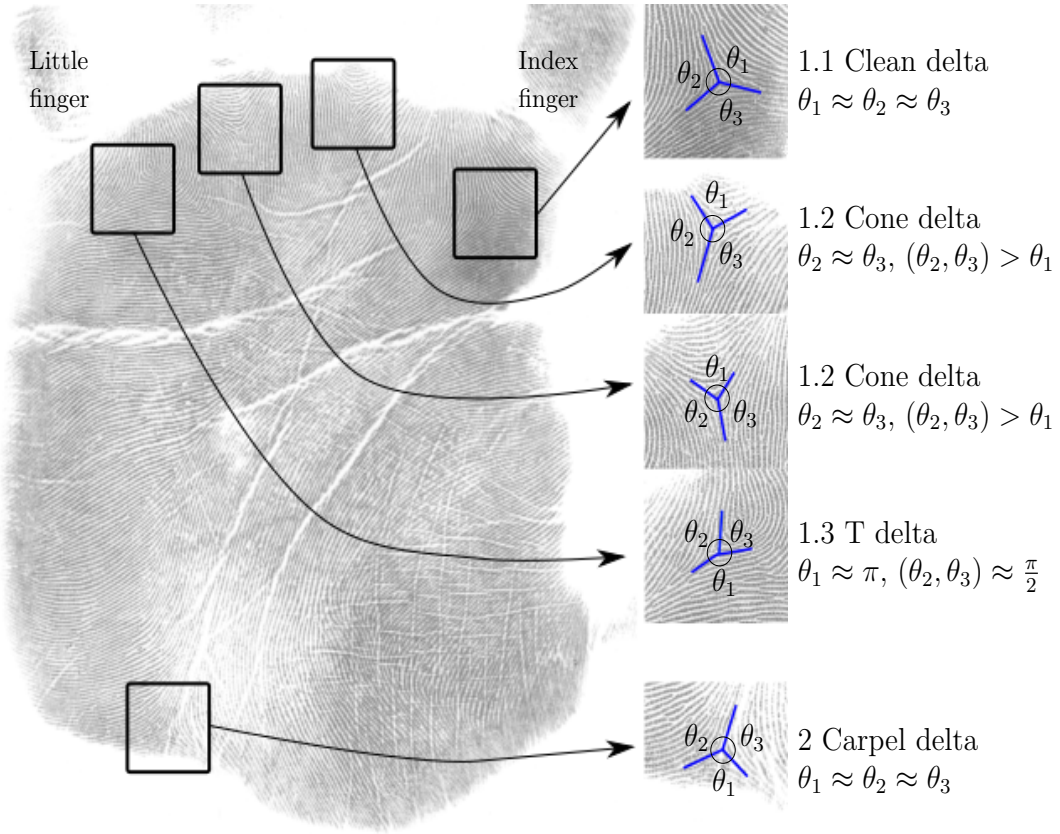
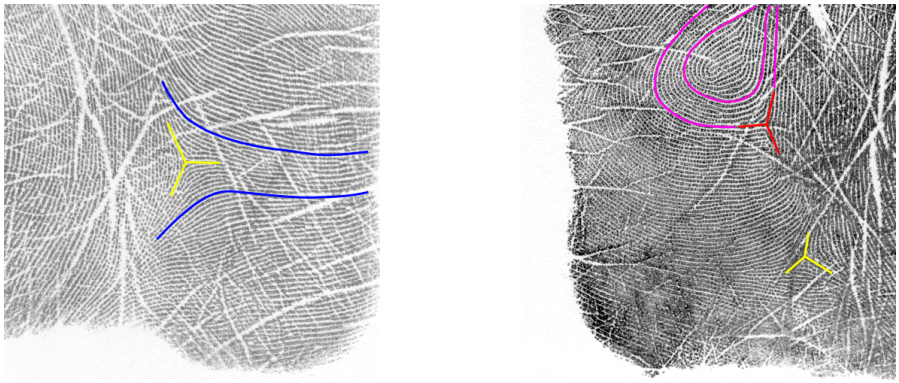


Figure 2.11.: Classifying typical deltas

Carpel delta The carpel delta is largely responsible for the level I attributes in the hypothenar region of the palm (which was discussed in section 2.2.2). When a lower carpel delta exists the angles which the friction ridges make with the centre of the delta are approximately equal. There are two further points to make about the carpel delta. Firstly, the carpel delta exists where there are many flexion creases, distinguishing it from a clean delta. Secondly, for the case when the carpel delta exists at a higher position within the palm, there is great variability in the angles subtended by the friction ridges and thus its difficult to generalise a typical case. Carpel deltas exists in more than 99.4% of palms [18].



(a) Funnel exiting the palm (b) Two Deltas and a Loop

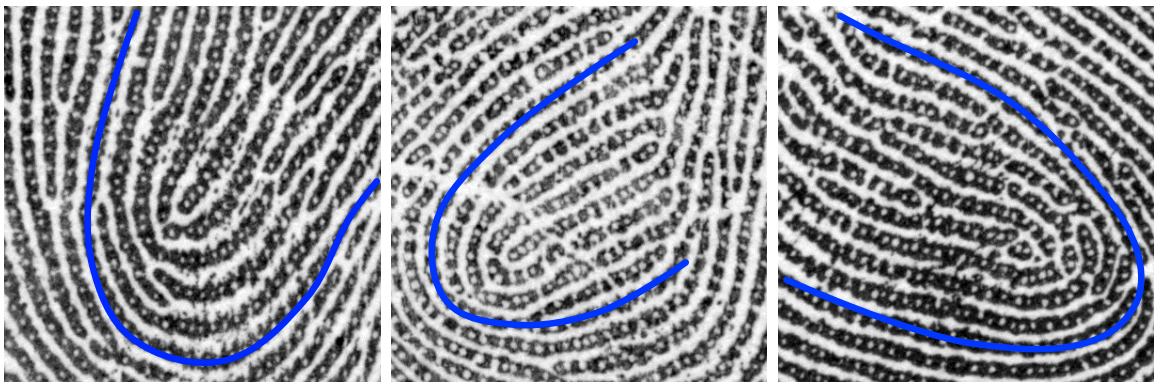
Figure 2.12.: Structures and ridge flow in the hypothenar, carpel delta (yellow), horizontal funnel (blue), extra delta (red) and connected loop (purple).

When the carpel delta is at a position higher than normal, this usually creates a funnel that exits the palm opposite the thumb, this is illustrated in 2.12 (b), high carpel delta and funnel are yellow and blue respectively.

Occasionally two deltas can be found in this region, where one is the carpel delta and the other is usually connected to a loop or whorl. This is illustrated in 2.12 where the carpel delta is yellow, the second delta is red and the connected loop is in purple.

2.3.2. Loops

A *loop* has a similar structure to a whorl except the friction ridges flow in to and out of the structure at a similar orientation. Loops are illustrated in figure 2.13. The friction ridges flow into the structure, form a loop about the centre, which then flow out of the structure in a similar angle to that of its entry. Loops can be further classified by the approximate average angle at which the friction ridges flow out of the loop. 2.16 (a), (b) and (c) illustrates loops denoted one, two and ten o'clock, respectively.



(a) (b) (c)

Figure 2.13.: Loops, where the ends of the blue line indicates their classification. (a), (b) and (c) illustrates loops denoted one, two and ten o'clock, respectively.

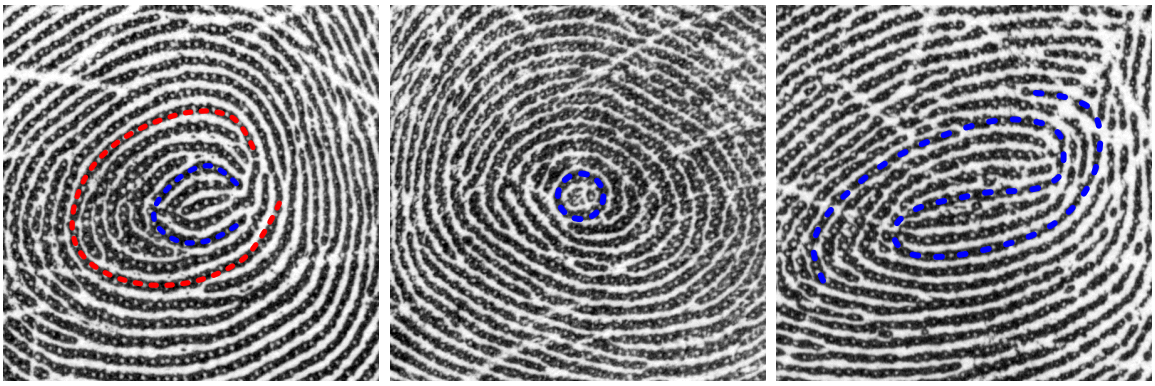
2.3.3. Whorls

A *whorl* is an approximately circular flow of friction ridges that are concentric about the centre of the structure. Whorls are illustrated in figure 2.14. The majority of the whorl friction ridges form a circuit, or a spiral about the centre of the structure. Unlike the delta, the whorl ridge flow is approximately symmetric about the centre of the structure. Whorls can be further classified using fingerprint terminology (Note: this terminology is not used in the palm print literature) into four categories: *plain whorl*, *central pocket loop*, *double loop* and *accidental whorl*.

Plain Whorl A plain whorl in fingerprint analysis is defined as having two deltas and one friction ridge making a complete circuit [10]. However, since two deltas need not exist near a loop on a palmprint (but must on a fingerprint), the require for two deltas on a palm print is unnecessary. Figure 2.14 (b) illustrates a plain whorl where the ridge completing a full circle is illustrated in blue.

Central Pocket Loop A central pocket loop is defined as being a structure that contains a combination of both a loop and a whorl [10]. An example of a central pocket loop is illustrated in 2.14 (a), where the loop and the whorl are illustrated in blue and red, respectively.

Double Loop A double loop is defined as being a structure that contains two separate and distinct loops, that either overlap or surround one another [10]. An example of a double loop that surrounds one another is illustrated in blue in figure 2.14 (c).



(a) Central pocket loop, where the loop and whorl are illustrated by the blue and red lines, respectively.

(b) Plain whorl, where the blue illustrates a friction ridge which completes a full circuit.

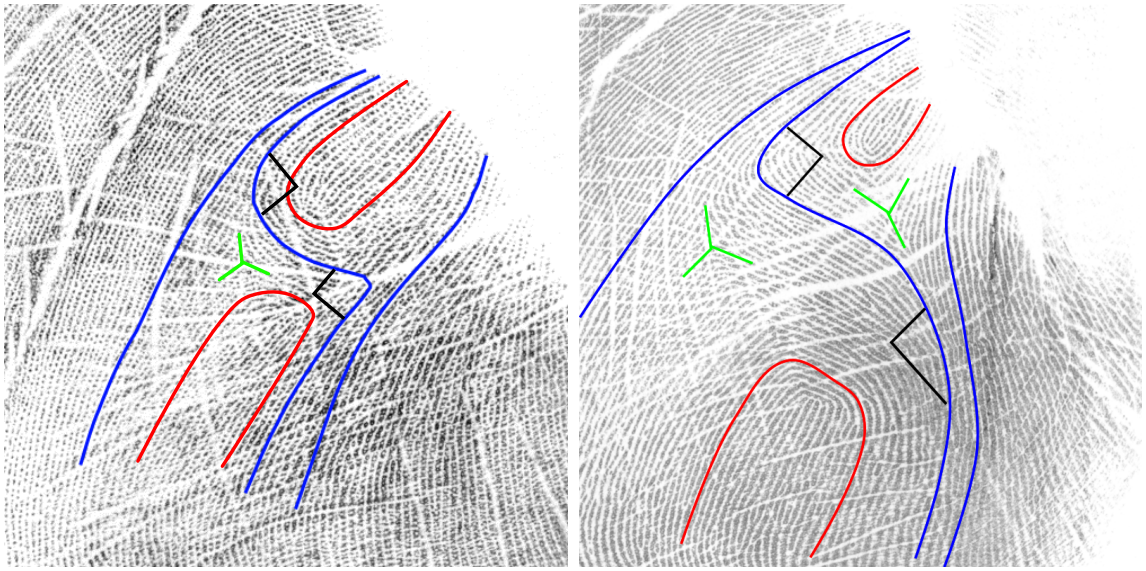
(c) Double loop, where the blue line illustrates two loops surrounding one another.

Figure 2.14.: Different types of whorls

2.3.4. Vestiges

A *vestige* is a type of Level I structure which is unique to the palm and also unique to the thenar region. Vestige structures can be considered as a special configuration of delta(s) and loops, in which one or more deltas lie approximately between two opposing loops. The specific structures contained within vestiges often varies, which make them difficult to define concisely [8].

Figure 2.15 (a) and (b) illustrates two different types of vestige, in which there are one and two deltas, respectively. The blue lines illustrate the main structure of the vestiges, where generally the centre blue line has two sharp angles (approximately 90 degrees), illustrated by the black lines. The two opposing loops are illustrated by the red lines. If the opposing loops are of a similar size and shape, the vestige is relatively symmetric; this is the case in figure 2.15 (a). However if the opposing loops have different sizes and shapes, the friction ridge flow of the vestige will vary and the two right angle may less apparent; this is the case in figure 2.15 (b).

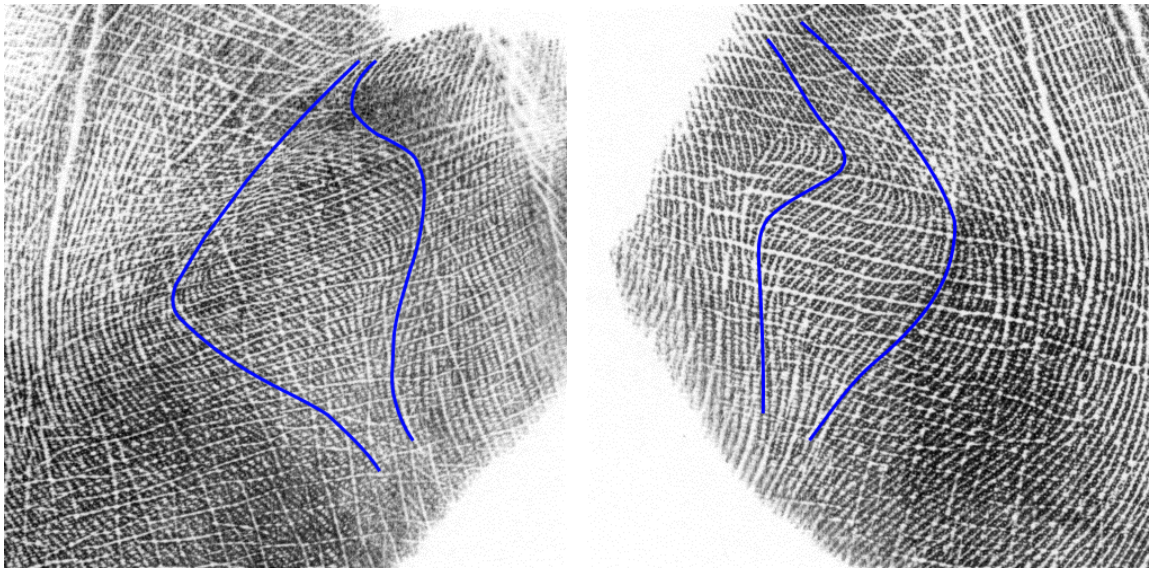


(a) A vestige with similar shaped loops.

(b) A vestige with different loops.

Figure 2.15.: Different forms of vestige, the main structure is illustrated by the blue lines, the sharp angles are illustrated by the black lines, the two loops are shown in red and the delta(s) in green.

A partial vestige has a single sharp angle and a and relatively smooth angle on opposing sides of the structure. Partial vestiges are illustrated in figure 2.16.



(a) Partial Vestige (b) Partial Vestige

Figure 2.16.: Partial vestiges (b) and (c).

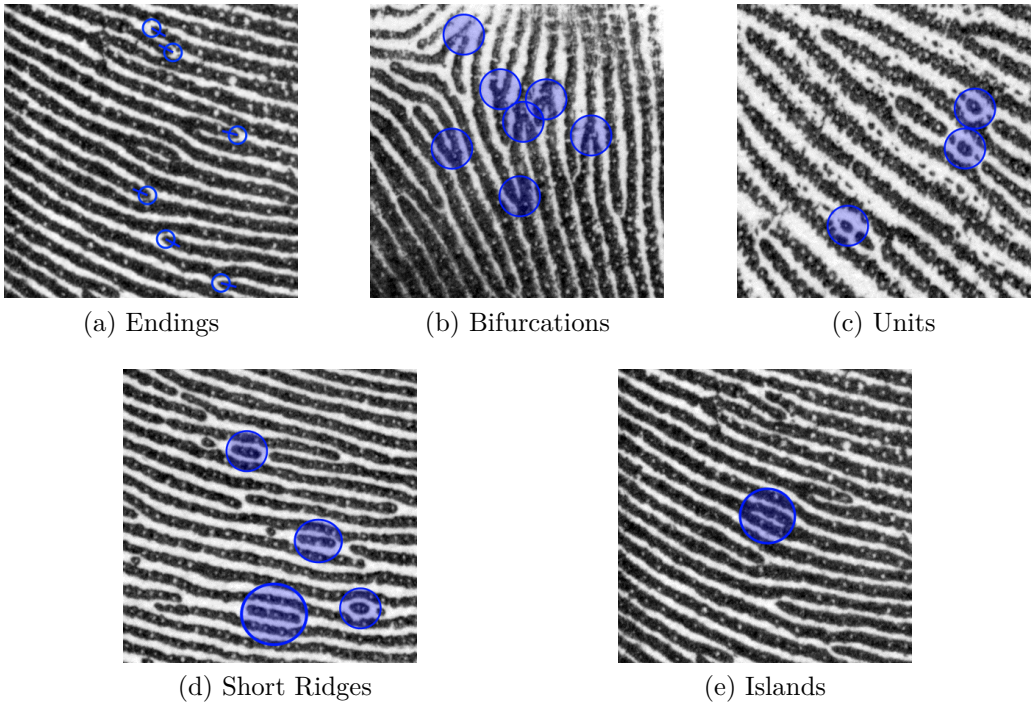
If the vestige has a square like ridge flow, it is known as a square nosed loop, the existence of square nosed loops is only found in the thenar region of palms.

If multiple vestiges or square nosed loops exist in the thenar, the length of the loop is longer at the base of the palm. [18] states that 6.9% of palms contain vestiges or partial vestiges, when no vestigial structures occur, loops/deltas and whorls exist in 1.15% and 0.2% of the dataset, respectively.

2.3.5. Level II Detail: Minutiae Points

There are particular properties of friction ridges which define them as *minutiae points* (also known as Galton detail [19]). Minutiae points include

1. **Friction ridge endings** - Where a single friction ridge exists, there must to be a start and an end point, these points are known as *endings* and are illustrated in figure 2.17a. These points generally have an associated position and an angle. The angle is defined to be the angle the friction ridge makes with the end point and the direction is illustrated in green in 2.17a.
2. **Bifurcations** - Any friction ridge which forks into two or more friction ridges, is known as a bifurcation, as illustrated in figure 2.17b.
3. **Ridge units** - A single *friction ridge unit* is defined as a *minutiae point*, as illustrated in figure 2.17c.
4. **Short friction ridges** - A small section of friction ridge is also defined as a minutiae points,as illustrated in 2.17d.
5. **Friction ridge islands** - Friction ridge islands occur when a friction ridge bifurcates into two friction ridges and then merges back into one friction ridge, as illustrated in figure 2.17e.



(a) Endings (b) Bifurcations (c) Units

(d) Short Ridges (e) Islands

Figure 2.17.: Minutiae points

2.3.6. Level III Detail

Level III detail is defined as *pores*, *edges, inter ridge detail*, *flexion creases, secondary creases* and *scars*. Generally, pores and inter ridge detail are only evident in the image when the palmprint resolution is 500 *ppi* (*Pixels Per Inch*) or above.

The following subsections describe and illustrate level III detail.

2.3.6.1. Pores

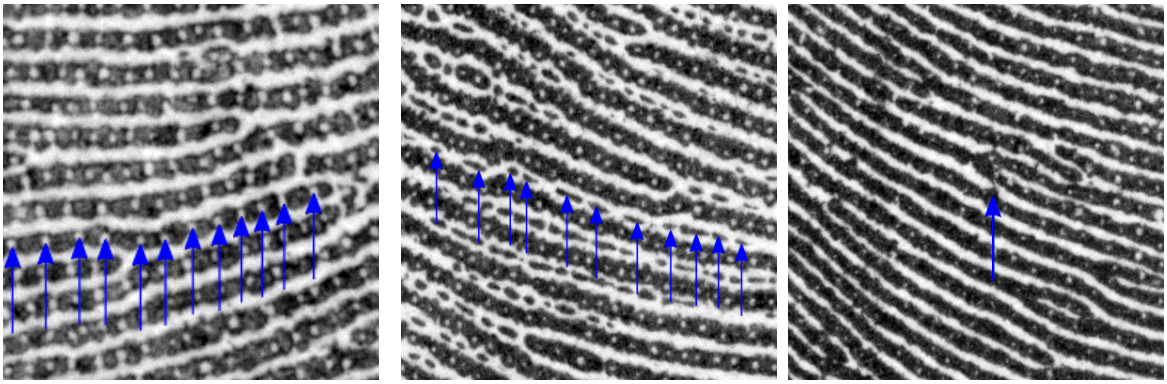
A *pore* is the opening in the surface of the epidermis of a sweat gland. Pores exist along the tops of friction ridges. Pores generally appear at regular intervals along the friction ridge, where the interval is approximately proportional to the breadth of the friction ridge[8]. As previously stated, a friction ridge unit is defined as a single section of friction ridge containing a pore. Pores along one friction ridge, are illustrated in 2.18 (a). The study of pores is known as poroscopy.

2.3.6.2. Edgeoscopy and Inter Ridge Detail

Edgeoscopy, is the study of the characteristics of the edges of the friction ridges [19]. Since the palmprints acquisition process can occasionally obscure or make it difficult to use Level I and II detail, edgeoscopy and poroscopy were developed to aid in the identification process. Since latent palmprints are generally a lot smaller than an exemplar palmprints, edgeoscopy and poroscopy could be very useful for extracting reliable discriminative features but are currently used.

Any aberrant structures that exist in between adjacent friction ridges are known as *inter ridge detail*. There are two distinctive types of inter ridge detail, namely, *incipient ridges* and *ridge edge protrusions*.

Occasionally, in the early stages of friction ridge formation, a friction ridge does not *fully* mature. These immature friction ridges are known as *incipient ridges* [8] and a



(a) Pores (b) Incipient Ridges (c) Ridge Edge Protrusions

Figure 2.18.: Level III detail, where the blue arrows illustrate pores, incipient ridges and ridge edge protrusions, in (a), (b) and (c), respectively.

set of incipient ridges along one valley, is illustrated in 2.18 (b). When an incipient ridge is compared with a fully matured friction ridge, the incipient ridge is a broken line.

Occasionally the epidermis on a volar pad can have aberrations. This causes aberrant formation of the the friction ridges. In this instance ridge units form, but do not fuse together in the normal way, leading to very short ridge units. This condition is known as Dysplasia, believed to be caused by a genetic malfunction [8].

When examining friction ridges at high resolutions (i.e greater than 500 ppi), it is apparent that friction ridges are not perfect ridge/valley structures. Occasionally, protrusions exist along the paths of friction ridges, known as *ridge edge protrusions*. An example of a ridge edge protrusion is illustrated in 2.18 (c).

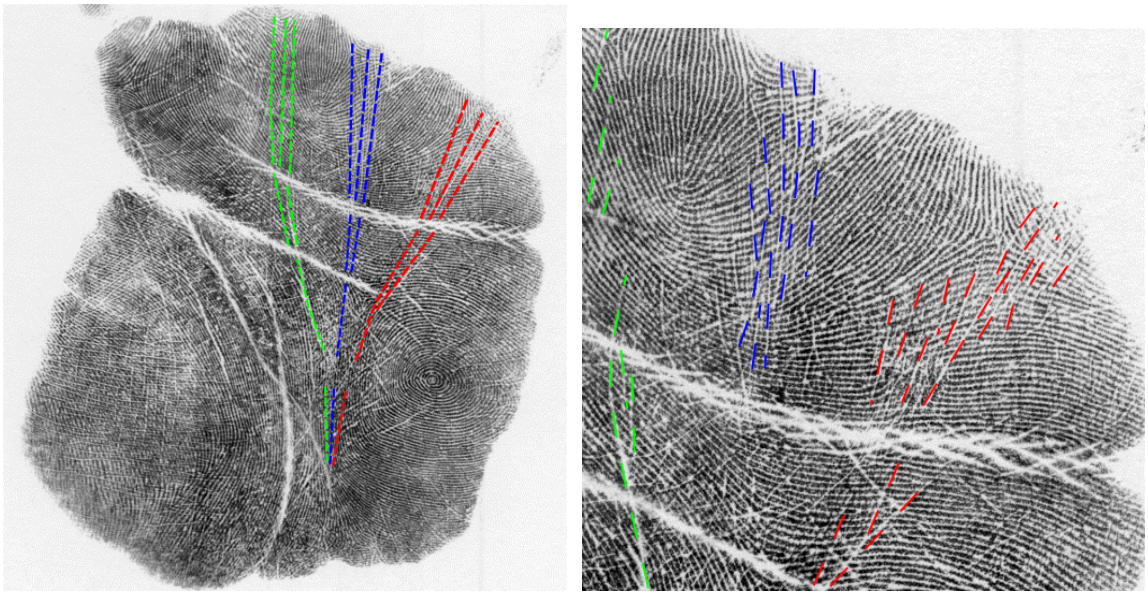
2.3.6.3. Secondary/Minor Creases

In a similar manner to major flexion creases, the deformation in the skin and muscle of the palm, create what are known as *minor* and *secondary* flexion creases. Minor and secondary flexion creases are shorter in length and width than major flexion creases, but generally they are sufficiently long so their trend can be used to distinguish them [15].

Minor Flexion Creases The minor flexion creases are further divided into four subgroups: *minor finger creases*, *accessory distal transverse crease*, *E lines* and the *hypothenar crease [15].*

Minor Finger Creases A set of secondary creases known as *minor finger creases,* span the hypothenar and the tri-radiate regions [8]. The minor finger creases start from the base of the hypothenar and extends into the tri-radiate, ending at the base of the second, ring and little finger, illustrated in green, blue and red in 2.19a, respectively.

Minor finger creases, generally are the thinnest creases on the palm. Minor finger creases do not exist in all palms, but when they do exist, they are very prominent. Of the literature reviewed within this report, none investigates the properties and frequency of minor finger creases as a means to use for identification purposes.



(a) General position of minor finger creases (b) Minor finger creases

Figure 2.19.: Minor finger creases

Accessory Distal Transverse Crease The *accessory distal transverse crease*, is a crease approximately parallel to the distal transverse crease and is illustrated in blue in 2.20. The accessory distal transverse crease exists in the tri-radiate, in between the distal transverse and the base of the fingers.

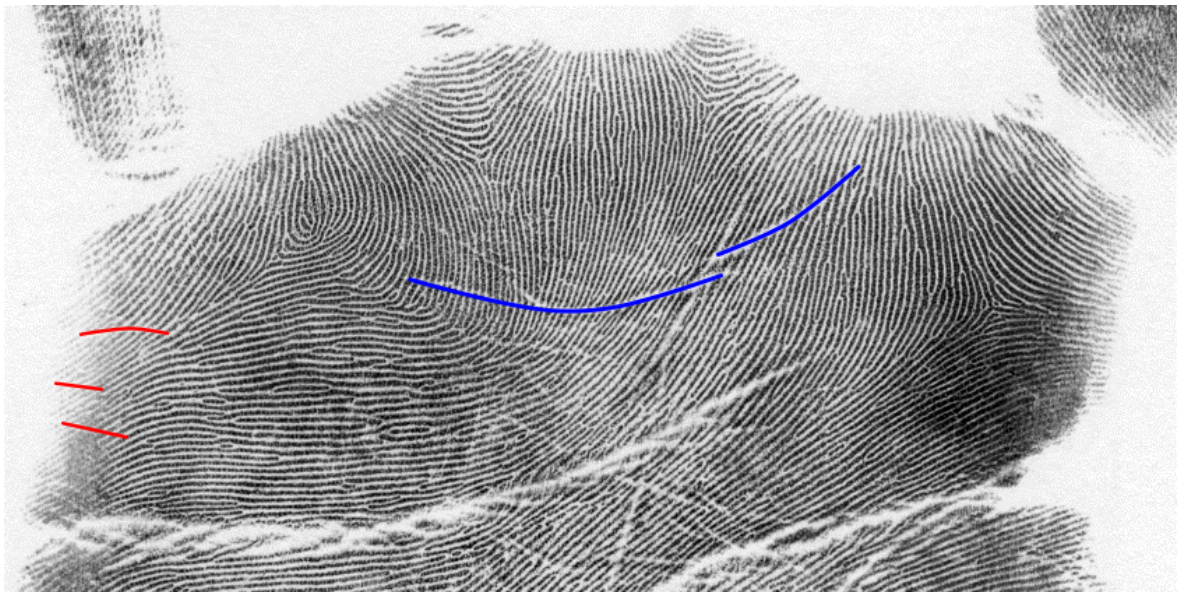


Figure 2.20.: Accessory distal transverse crease (blue) and E lines (red).

E Lines Edge creases cut into the ridge flow, below the base of the little finger, as illustrated in red in figure 2.20. These edge creases are known as *E lines*, as they often resemble the horizontal components of a capital E.

Secondary Flexion Creases Any crease that is not classified as *major* or *minor* flexion crease is defined as a secondary flexion crease [15]. Secondary creases can exist in all of the regions of the palm.

The Tri-radiate Region As previously stated, the layer of subcutaneous fat beneath the tri-radiate allows deformations in the epidermis. These deformations can create a range of different creases.

The boundary at the top of the tri-radiate region is where the base of the finger attaches to the palm. There are usually creases at this boundary, below the base of each finger. These creases are formed by the deformation of the muscle and skin due to the movement of the fingers.

The Hypothenar Region In a similar manner to E lines, the hypothenar region also has *edge creases*. They enter the hand from the very edge of the palm, chopping through the flow of the ridges at their respective points.

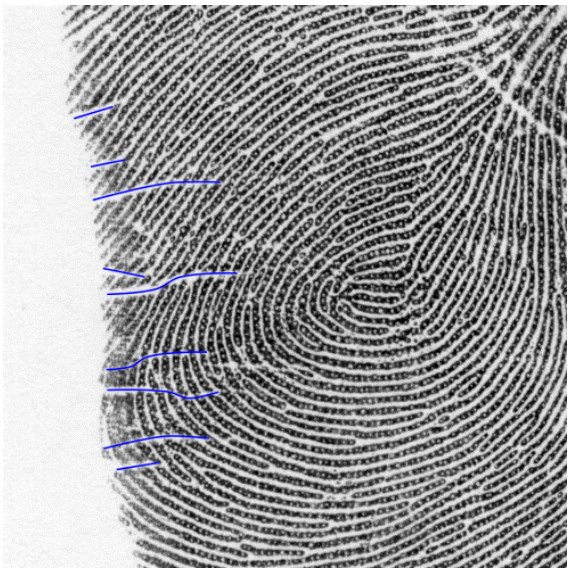
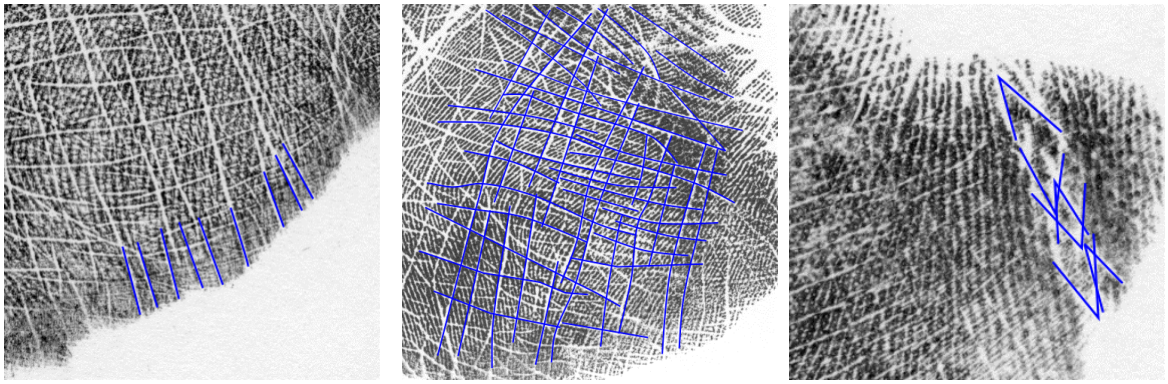


Figure 2.21.: Hypothenar edge creases, shown by the blue lines, chopping through the friction ridges.

The Thenar Region Thenar edge creases, usually enter the palm around the base of the thumb, approximately following the direction of the ridge flow at that point. An example of thenar edge creases is illustrated in 2.22 (a).

Thenar edge creases introduce one of the most distinctive features within the palm, known as *cross hatching*. An example of cross hatching is illustrated in 2.22 (b). Cross hatching is where two sets, of approximately orthogonal creases occur. Cross hatching exists only in the thenar region around the thumb. These cross hatches are created as a result of the movement and the flexibility of the thumb. They do not exist on the palm of a new born child and they tend to increase in length and breadth with age.



(a) Thenar edge creases (b) Cross hatching (c) Thumb bracelet crease

Figure 2.22.: Secondary flexion creases in the thenar region, where the ceases are illustrated in blue.

There is also a minor crease in this area at the point at which the thumb joins the thenar region; known as the *thumb bracelet crease*. It often resembles two sets of interlocking, opposing crows feet.An example of a thumb bracelet crease is illustrated in 2.22c.

Wrist Bracelet Crease The *wrist bracelet crease* is a set of interlocking v's that separates the base of the palm (the base of both the Thenar and Hypothenar regions) from the fore-arm. It represents the boundary for the base of the thenar and the hypothenar. There are no friction ridge formations below this crease. The crease has many discriminative features, but due to the fact it is both 1) not captured by livescan (the USA and UK Police force uses Live-scan, therefore the wrist bracelet crease is unavailable in their exemplar palmprints [7]) 2) rarely found at crime scenes.

Deterioration of the Flexion Creases The friction ridges and the epidermis tend to deteriorate with age [20]. This gives the effect of the flexion creases *growing*, as the individual ages. The effect of flexion crease growth, was illustrated by Sir William Herschel, who pioneered research in fingerprint identification, by taking prints of his left hand, thirty years apart. The two prints of Sir William Herschel's left palm are illustrated in figure 2.23.

Scars Under certain circumstances, scars can cause disruptions in the flow of friction ridges. A study of the use of permanent scars as features is presented in [21], the study concludes that the comparison of scars in friction ridges is perfectly valid. The epidermis does not necessarily heal in the same position after a cut. A scar is illustrated in 2.24.

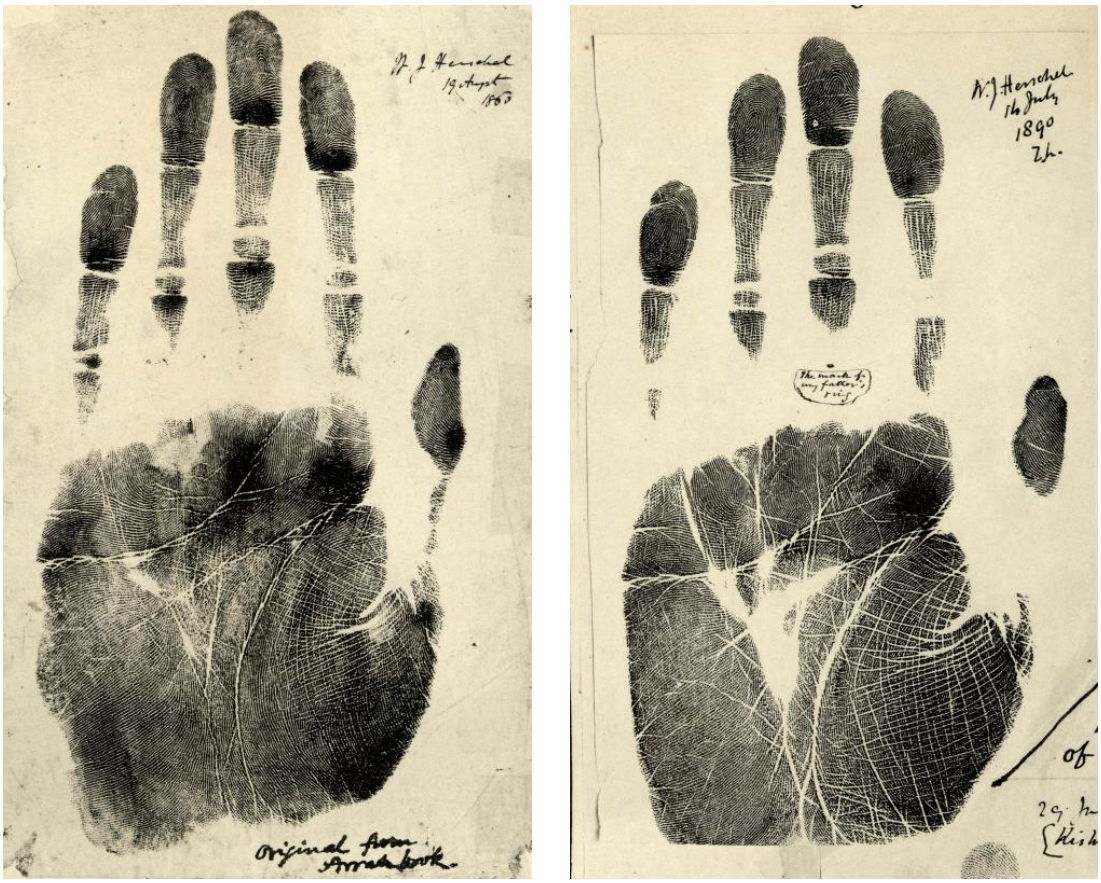


Figure 2.23.: Two prints of Sir William Herschel's left palm, the print on the right was taken thirty years after the print on the left, to show the effect of flexion crease growth. *By courtesy of The Curve, Slough Libraries* [a].

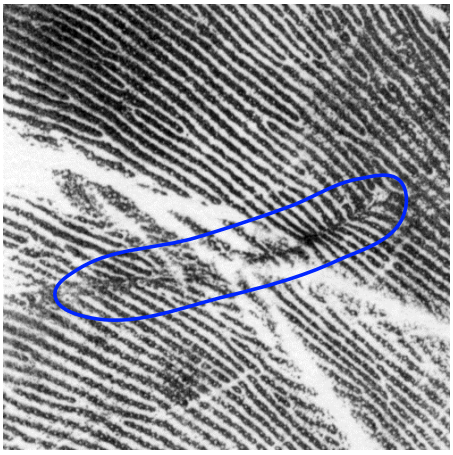


Figure 2.24.: Scar

Scars are very useful for identification purposes, if the exemplar was acquired post scaring [7].

2.4. Summary

The taxonomy of Level I, II and III detail of the palm [5, 12, 8, 10], closely follow the taxonomy finger prints. However, there is no general agreement on the exact

terminology; even different police forces use terms for the structures and detail [10, 5]. Figure 2.25 illustrates the Level detail taxonomy used within this thesis and that which is also commonly used by forensic investigators in the UK.

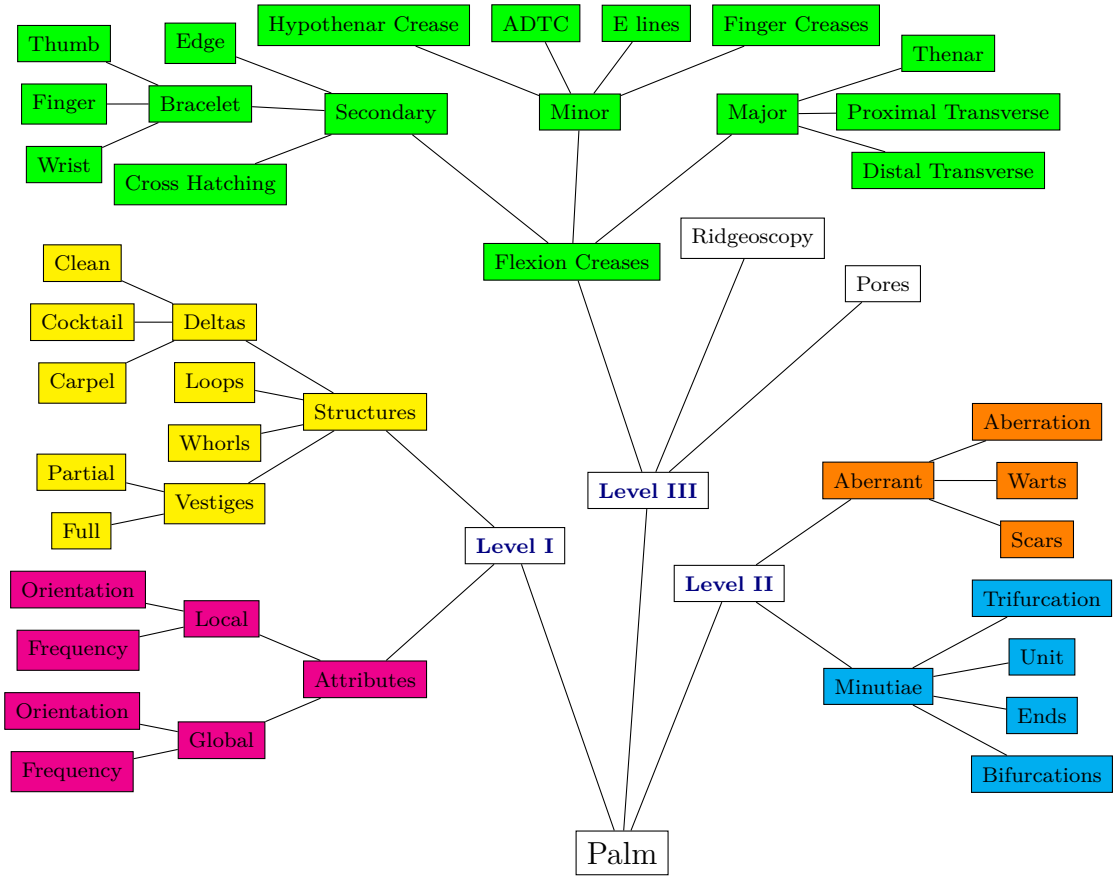


Figure 2.25.: A graphical representation of the detail of a palmprint

It should also be noted that few studies have investigated the frequency of particular Level I structures within the regions of the palm. 2.1 summaries the structures intrinsic to each region of a palm taken from [10, 18, 4].

Feature	Tri-Radiate	Hypothenar	Thenar
Deltas	*3+*	*1+*	*0 unless part of a vestige*
Whorls	*Yes*	*Yes*	*Yes*
Loops	1+	*Yes*	*Yes*
Vestiges	*No*	*No*	*Yes*
Funnels	*Small*	*Large 2+*	*None*
Other	*Often loops & whorls*	*Large area of ridges*	*Cross-hatches/scratches*

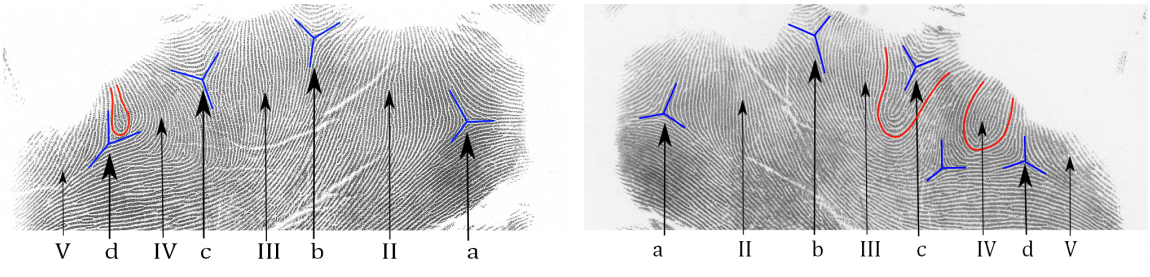
Table 2.1.: Intrinsic Level 1Features of the Major Regions

When considering the robustness of any matching process using the Level I, II, III detail the effect of age must be taken in to account. The flexion creases and the epidermis deteriorate grow and deteriate with age (see figure figure 2.23). Therefore, the finer detail (Level II and III detail) will change with time, as the flexion creases expand and widen. Conversely, since Level I detail (ridge flow) covers a large area of the palm

which varies slowly within a local area, the growth of the flexion creases will have a less influence upon Level I detail as the palm ages. Therefore, the use of Level I potentially, provides a more robust matching strategy.

Although current APIS exclusively use Level II detail for matching (i.e. minutiae), two attempts [18, 22] have been made to use Level I detail in the categorisation of the triradiate region of exemplar palmprints. In [18] the analysis was conducted on a database collected by fourteen Arizona Law Enforcement Agencies consisting of 998 left and right palms from 499 persons. The analysis began by extending the taxonomy of the triradiate to include the position and frequency of Level I structures. The extened taxonomy segmented the tri-radius into sub-regions using the orientation of the flow of the friction ridges [12].

The extended taxonomy defines new sub-regions, denoted as *tri-radiate pads*[1] *I, II, II, IV and V*, as illustrated in figure 2.26.



(a) **4**(IV) (b) **5**(III-IV,IV)

Figure 2.26.: Classifying the tri-radiate by its aberrant structures. Loops and delatas are illustrated in red and blue, respectively.

2.26 (a) illustrates the four *normal* deltas, denoted *a*, *b*, *c*, and *d* for the delta below the fore, second, ring and little fingers respectively.

The tri-radiate pads were used to *manually* classify the number and position of the loops and deltas. If any of the normal deltas are missing, the missing deltas position is also recorded. However due to the complex nature of the formulation of friction ridges, many different patterns exist [8] and therefore a significant amount of forensic expertise was required during the classification process.

As an example, the palm illustrated in 2.26 (a), is classified as **4**(IV) because since there are no missing deltas and only one loop occurs in position IV and in figure (b), the palm is is classified as **5**(III-IV, IV) since there is an extra delta (in IV) and two loops occur in position III-IV and IV. The study concluded with the observation that current APIS would be improved, if they were to include level 1 detail. However, it also concluded that specialised training would be required in order to perform such classifications.

2.5. Conclusions

The chapter has illustrated the variety of features which exist within Level I, II and II detail. It has noted in that current APIS only use a subset of Level II detail and

[1] *In the UK, the region is denoted tri-radiate. In the USA, the region is denoted intedigital. Hence, in the UK the pads should be denoted tri-radiate pads I,II, III, IV and V .*

that this subset is subject to deterioration and owing to the manual identification procedures, is often poorly recorded [1].

A small scale study presented in [1] shows that crime scene latent palmprints originate from each of the major regions of the palm. Approximately 40-45% of all crime scene latent palmprints originate from the hypothenar, 23-29% originate from the thenar and 26-37% originate from the tri-radiate regions. In [18] 95.2% of exemplar palm prints had at least four deltas in the tri-radiate. This information, which is available in both crime scene latent print and exemplars, is more robust to ageing but is currently unused in automatic systems. However, in order to exploit Level I information in [18], a more detailed taxonomy had to be developed and the manual identification was said to be laborious and requiring expert training.

Chapter 7 provides a method for the automatic identification of the Level 1 structure (i.e delta, loops and whorls). Chapter 8 provides a method for the automatic identification and classification of regions of Level I *detail*. Each of the methods relies upon an accurate estimation of the orientation field. An improved orientation estimate is provided in chapter 5

Chapter 3

APIS

3.1. Overview of Current APIS Processing

The general methodology of current APIS, is a modified form of the methodology used in fingerprint matching (i.e AFIS) [6]. The elements of the finger print methodology are modified to cope with the much larger variety of detail and the increased size of the palm [5]. For example, finger prints do not contain distinct regions or the same variety of flexion creases, and therefore, unmodified fingerprint techniques do not accurately extract reliable features [23]. The same general stages of the methodology are used for exemplar and latent print and are listed below.

1. **Enhancement**
   Given that current APIS only use on Level II detail, all other detail within the palm is regarded as noise, and therefore the function of this stage is the removal of flexion creases and all Level III detail from the print. This is achieved by the use of a series of contextual filters [24] applied locally to the palm print, where the parameters of the contextual filter are derived from the local friction ridge orientation and density. The individual stages within enhancement are shown below.

   a) *Pre-processing*

   b) *Friction ridge orientation estimation*

   c) *Friction ridge density estimation*

   d) *Contextual filtering*

2. **Feature Extraction**
   The enhanced image is thresholded to produce a binary image, which is then thinned to produce a skeleton. The skeleton is a 'line-drawing', of single pixel width, which locates the central path of the friction ridges. The skelotonised images are then processed to identify the locations of friction ridge end points and bifurcation (i.e minutiae) using Rutovitz's crossing number [25]. The resulting image is finally filtered to remove unreliable minutiae. The stages of feature extraction are shown below.

   a) Thresholding

   b) Thinning and skeletonisation.

   c) Minutiae extraction

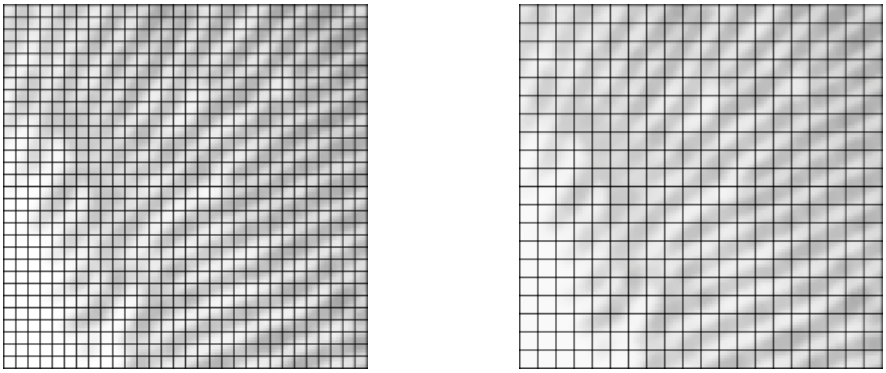
   d) Minutiae filtering

3. **Matching**
   Typically matching is performed between sets of minutiae and is typically done using minutiae cylinder code (MCC) [26] or variants thereof. The results of the matching process are a ranked list of potential matches from the database of exemplars, ranked according to a matching score.

The following sections follow the categorisation given above and describe the variants used within each of the individual stages.

3.2. Enhancement

As previously stated, the purpose of enhancement by contextual filter is to remove noise, flexion creases and all Level III detail from the original palmprint, leaving only the main structure of the friction ridges. In the majority of cases (for example [27, 28]), a palmprint image is divided into blocks of size n by n pixels and then local attributes (such as orientation and density) are estimated for each block (i.e steps 1(b) and 1(c) in section 3.1). The size of the blocks differ from method to method. Figure 3.1 shows 4 by 4 and 6 by 6 blocks overlaid upon a palm print to illustrate the scale of the detail which is analysed.



(a) 4 by 4 blocks (b) 6 by 6 blocks

Figure 3.1.: Segmenting a palmprint image into blocks.

For each block within the palmprint, orientation and density estimates of the local friction ridges are calculated. Contextual filters are created using the local orientation and density estimates. Each block is then filtered by its contextual filter to enhance the individual block. The following subsections describe the techniques which have been used in each of the four steps of the enhancement.

3.2.1. Pre-processing

The most common pre-processing technique is to first filter the image with 3 by 3 Gaussian filter and follow this with a 3 by 3 median filter [5]. However, little or no rationale has been found for the use of the approach and there is no agreement in the literature as to which value of standard deviation σ is to be used in the Gaussian filter (it is often left unspecified as in [27]).

3.2.2. Friction Ridge Orientation

Estimation of the friction ridge orientation field is the most critical step in the enhancement process, as both the frequency of the friction ridges and the Gabor filter are guided by the friction ridge orientation estimate.

Friction ridge orientation estimation techniques can be split into three general approaches : *slit-based*, *frequency domain* and *gradient-based* methods.

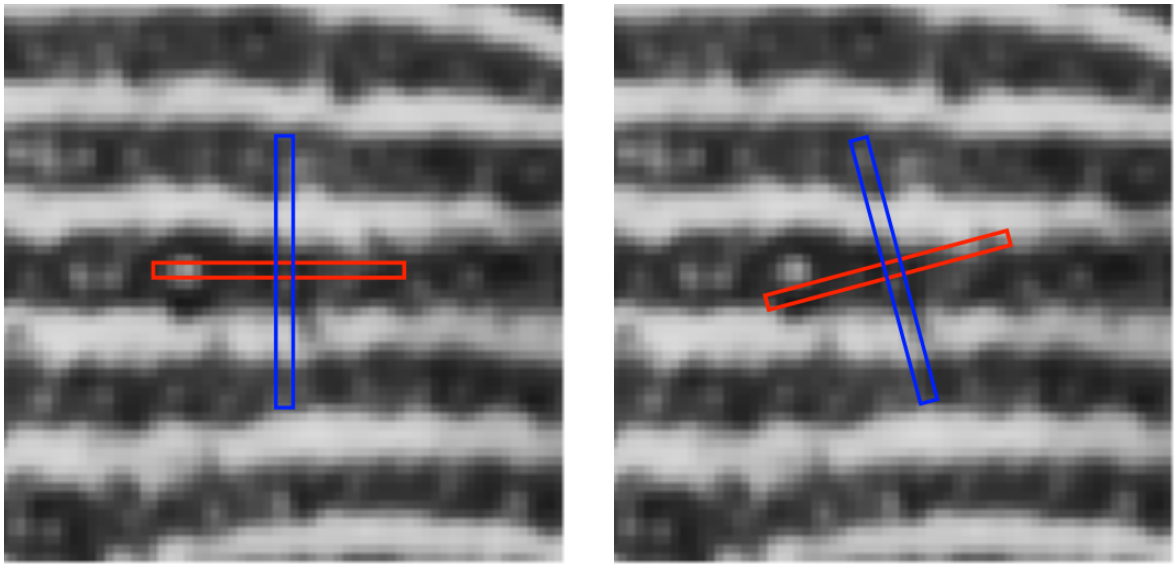
Slit and frequency based methods calculate an aggregate orientation estimate for non-overlapping local blocks within the image. Therfore, all pixels within a block have the same orientation estimate. Gradient based methods on the other hand calculate an orientation estimate for each pixel, often using a sliding window.

Methods for comparing the accuracy of friction ridge orientation extraction techniques on fingerprints have been developed [29, 30], however to date no similar methods have been developed for palmprint friction ridge orientation. Section 5.3 in chapter 5 presents a new method for assessing the accuracy of the orientation estimate generated by different techniques.

3.2.2.1. Slit Based Methods of Friction Ridge Orientation

Slit based methods [28, 31] compare the grey level values along a pair of orthogonal slits, located at the centre of a block, as illustrated in figure 3.2. The slits are rotated a fixed number of times about the central point (as shown in figure 3.2) and a value of contrast is calculated for each pair of slits. The orientation of the pair of slits which has the highest contrast is then assigned as the orientation for that block. The method proposed in [32] chooses the orientation based on the largest difference between the standard deviation of the orthogonal slits.

A number of slit based methods have been applied to palmprints [28, 31, 33, 34].



(a) Slits at 0 and $\pi/2$ (b) Slits at $\pi/5$ and $6\pi/10$

Figure 3.2.: Orthogonal slits at the centre of part of a palmprint image

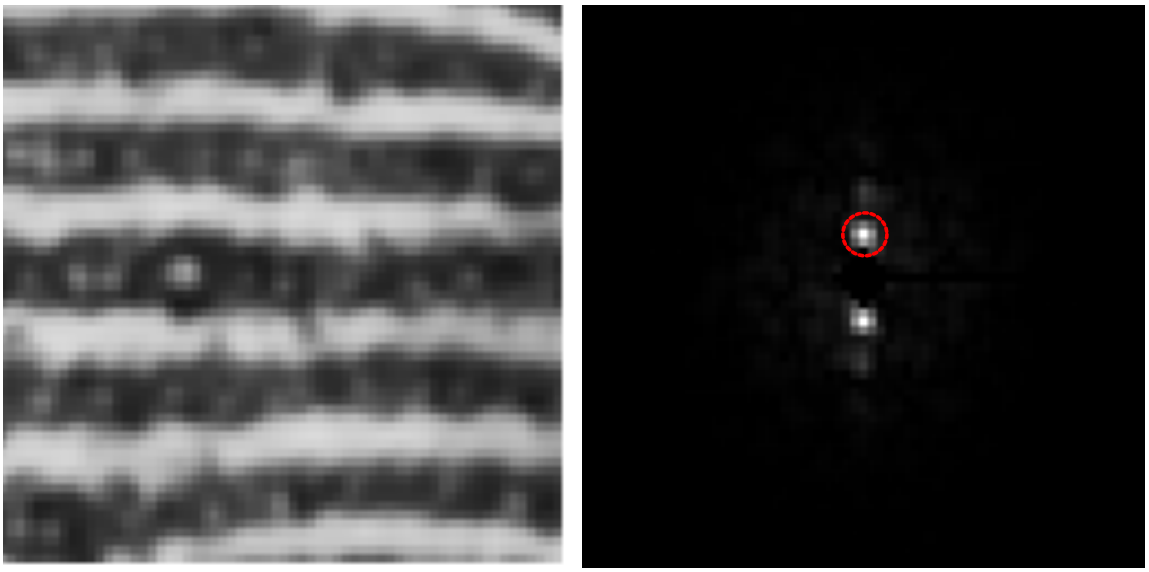
The method proposed in [31] post filters the friction ridge orientation to improve the orientation estimate in what it calls *special areas* (i.e. Level I structures). However, the interest in the slit based methods seems to have cooled, possibly due to the results from [29] who compared the accuracy of slit, gradient and frequency method for fingerprints. The study highlights two disadvantages. Firstly, the estimate produced by slit based methods are by their nature discrete and the associated quantisation error is determined by the angle between the rotations. Secondly, the computational complexity of slit-based methods is higher than the other approaches and proportional to the number of blocks.

3.2.2.2. Frequency Domain Based Methods of Friction Ridge Orientation

Frequency domain based methods split the palmprint image into non-overlapping blocks (of size M by M pixels) and perform the discrete Fourier transform (DFT) on each block to determine the main orientation in each block. They exploit circularly property of the DFT [35] which extends finite length spatial signals into the frequency

domain using the superposition principal [36]. The dominant orientation in the block is said to be the angle of rotation of the peak with the maximum amplitude in the frequency domain (when measured with respect to the centre of the DCT). This principal is illustrated in figure 3.3, where figure 3.3 (a) illustrates a palmprint block and 3.3 (b) illustrates the absolute value of the DFT of the palmprint block and one of the peaks is circled in red.

Frequency domain based method were first introduced in order to estimate the orientation of fingerprint friction ridges [37]. This was later adapted to model friction ridge flow using Short Time Fourier analysis, STFT analysis [38]. A method, which built upon the result of [38] but is specific to palms, was proposed in [39]. This method uses a region growing technique to reduce the effect of flexion creases and is used as a comparison for the methods in this thesis in chapter 5.



(a) A 64 by 64 Block

(b) DFT- Absolute value of the frequency domain of (a)

Figure 3.3.: Example of the Frequency domain based method - (a) is an block extracted from a palmprint image, (b) is the absolute value of the DFT of (a)

Frequency domain based methods can result in more than one orientation for a block, if all of the peaks in the frequency domain are considered. By exploiting this possibility, [40] reports a method to identify the separate orientation estimates of overlapping fingerprints.

One of the disadvantages of the frequency domain based method is the high computational complexity [6], which arises from both DFT and the method used to identify the peaks. In addition, in common with slit-based methods, the orientation estimate is discrete.

3.2.2.3. Gradient Based Methods of Friction Ridge Orientation

The most intuitive approach to the calculation of the orientation estimate is the use of a gradient based method, in which the rates of change in both the x and y directions are first calculated. The final orientation estimate is then calculated by applying atan2 to the rates of change to provide an estimate which lies within the $-\pi$ to π range. Gradient based methods are considered to be the most accurate form of estimate when

applied to good quality palmprints [29]. However, the approach is less accurate at the peaks and valleys of the friction ridges and in areas of low rates of change [6, 41, 29]. So called Adaptive gradient based methods [27, 42], have been developed to tackle these problems and are becoming a standard method for fingerprint friction ridge orientation estimation. These techniques have improved performance in lower quality finger prints and the have been applied to palmprints [41]. A detailed comparison between such techniques and those created in this thesis are given in Chapters 5 and 7.

3.2.3. Friction Ridge Density Estimation

The friction ridge density (sometimes called frequency, f) is the number of pixels (orthogonal to the friction ridge orientation) a friction ridge requires to complete a full cycle. The friction ridge density is illustrated by the white arrows in figure 3.4.

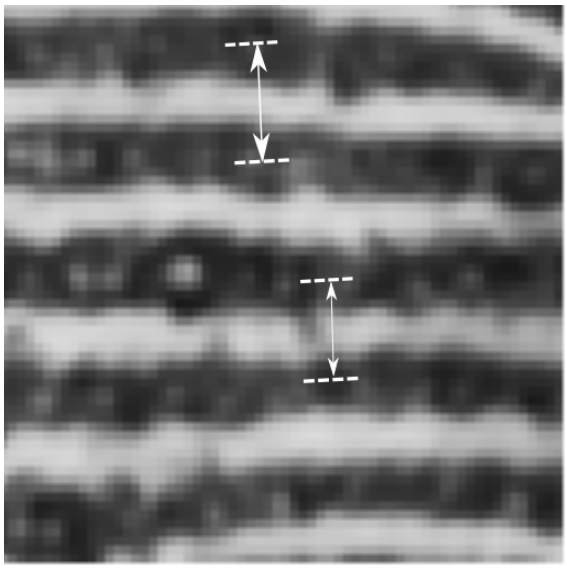


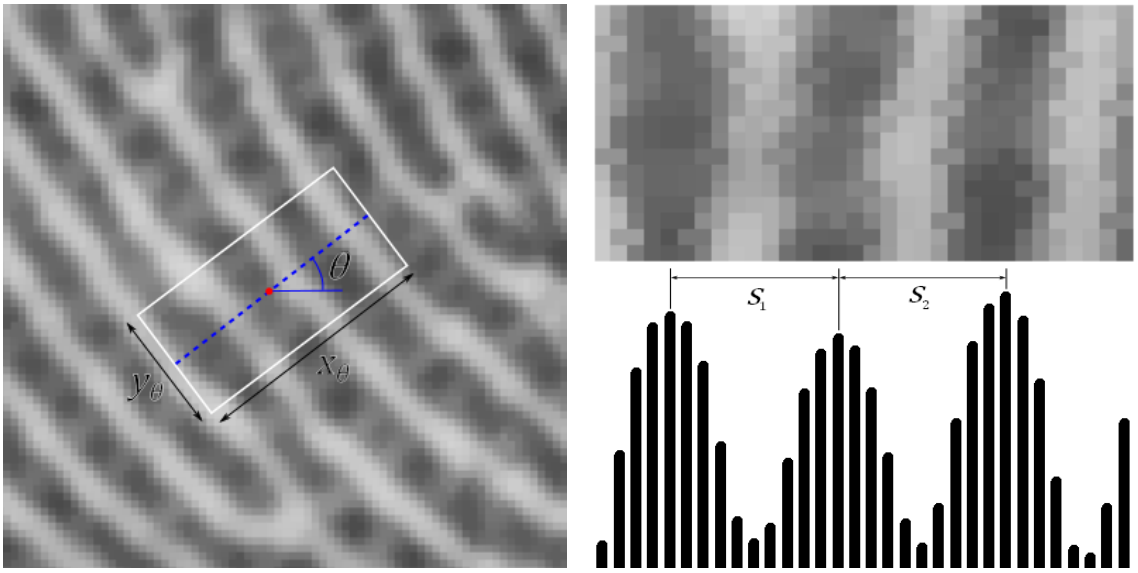
Figure 3.4.: Friction ridge density, dotted lines indicate the top of the friction ridges and the arrow illustrate the length of the friction ridge cycle

There are two fundamental approaches to the way in which ridge density field is calculated, these approaches are *spatial domain* and *frequency domain* methods. Spatial domain methods analyse the local ridge/valley features of friction ridges within the image, whereas frequency domain methods, transform the local image into the frequency domain and then calculate the result using the distance to the maximum peak.

In the spatial domain approach, the orientation of the friction ridges in the local area is required prior to extracting the friction ridge density. The method implemented in [43] (illustrated in 3.5(a)) uses an orientated window (white) of size 32 by 16 (x_θ and y_θ), at an angle (θ) orthogonal (dotted blue) to the direction of the ridge flow. This oriented window has been extracted from the image and is shown in the top of 3.5 (b). The *x-signature* is defined to be the sum of the columns of the oriented window, and is illustrated as a bar chart at the bottom of 3.5 (b).

The x-signature is the average of the ridge/valley ridge structure at 16 points, following the direction of the ridges. If no minutiae or structures exist within the oriented window, the average peak to peak length is said to be the density of the friction ridges. Therefore, ridge density is calculated as follows.

Let m equal the number of peaks in the x-signature, then the ridge density for that



(a) Orientated Window (white) (b) x-signature

Figure 3.5.: Extracting the ridge density using the spatial domain method.

particular orientated window , $n = m - 1$, where, s_i is the distance measure in pixels between the i^{th} and the $(i + 1)^{th}$ peaks.

$$f = \frac{n}{\sum_{i=1}^{n} s_i}$$

There are variations of the x-signature method,for example [27] uses a window size of 48 by 16, it then smooths the x-signature over a window size of 3 by local averaging and then estimates the ridge frequency by taking the median of the x-signature . The x-signature can also be modelled as a one-dimensional sine wave as in [6], where the friction ridge frequency is taken to be the frequency of the modelled sine wave.

The frequency domain method of friction ridge density estimation is calculated at the same time as the orientation estimate described in section 3.2.2.2. The method is illustrated in figure 3.3 and uses the distance between the peak and the centre of the FFT as an estimate of friction ridge density. It is known that the frequency domain method is more immune to noise, but the spatial domain method is more efficient to implement [6].

3.2.4. Contextual Filtering

This is the stage of an APIS/AFIS where the enhancement of the image is performed, this stage is often also referred to as *contextual filtering* [6].

To perform contextual filtering, the friction ridge orientation (section 3.2.2) and the friction ridge density (section 3.2.3) for each block of the palmprint image must first be estimated. A filter is then created for each block using these attributes, finally convolution is then performed resulting in the corresponding block of the enhanced image. This overall process is often known as *block by block filtering*.

The method described in [44] was the first method to give a solid rationale to the concept of block by block filtering. This method [44] uses a filter which is of a similar

shape to that of a friction ridge. The method proposed in [43] builds upon the concept of block by block filtering using an *even-symmetric Gabor filte*r to enhance each block. Contextual filtering using Gabor filters [43] has become a widely used method in both APIS and AFIS.

An even-symmetric Gabor filter is defined as:

$$G(f,\theta) = \exp\left\{-\frac{1}{2}\left(\frac{x_\phi^2}{\sigma_x^2} + \frac{y_\phi^2}{\sigma_y^2}\right)\right\}\cos\left(2\pi f x_\phi\right) \tag{3.1}$$

$$x_\phi = u\cos\theta + v\sin\theta \tag{3.2}$$

$$y_\phi = -u\sin\theta + v\cos\theta \tag{3.3}$$

Where σ_x and σ_y are the standard deviation of the Gaussian envelope in the direction of the filter and the orthogonal direction of the filter respectively, u and v are the coordinates in the frequency domain such u, $v \in \mathbb{R}^2$; and where f is the friction ridge density (calculated using the methods in section 3.2.3) and θ is the orientation of the friction ridges for each block (calculated using the methods in section 3.2.2). An even-symmetric Gabor filter is illustrated in figure 3.6, where $\theta = \frac{\pi}{4}$ and $f = 0.1$.

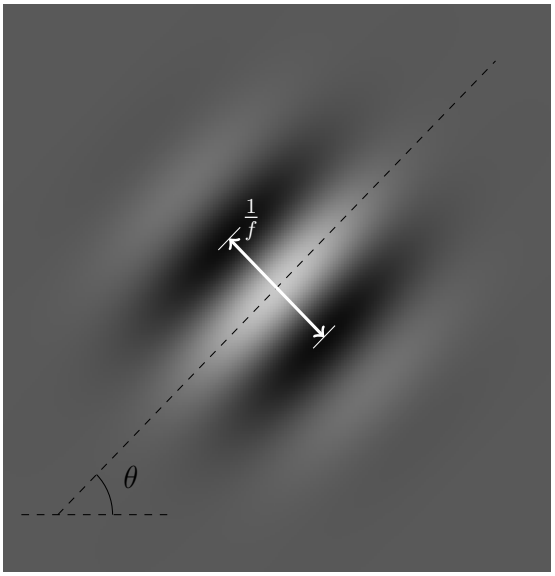


Figure 3.6.: An even-symmetric Gabor filter. The angle is illustrated by the black dotted line and the frequency is illustrated by the white arrow.

To reduce the computational complexity involved in creation of the filters during processing, [43] suggests using a pre-calculated bank of filters and choosing the Gabor filter from the filterbank which best matches the attributes of the local block. Figure 3.7 illustrates a filterbank of even-symmetric Gabor filters, for a range of f and θ.

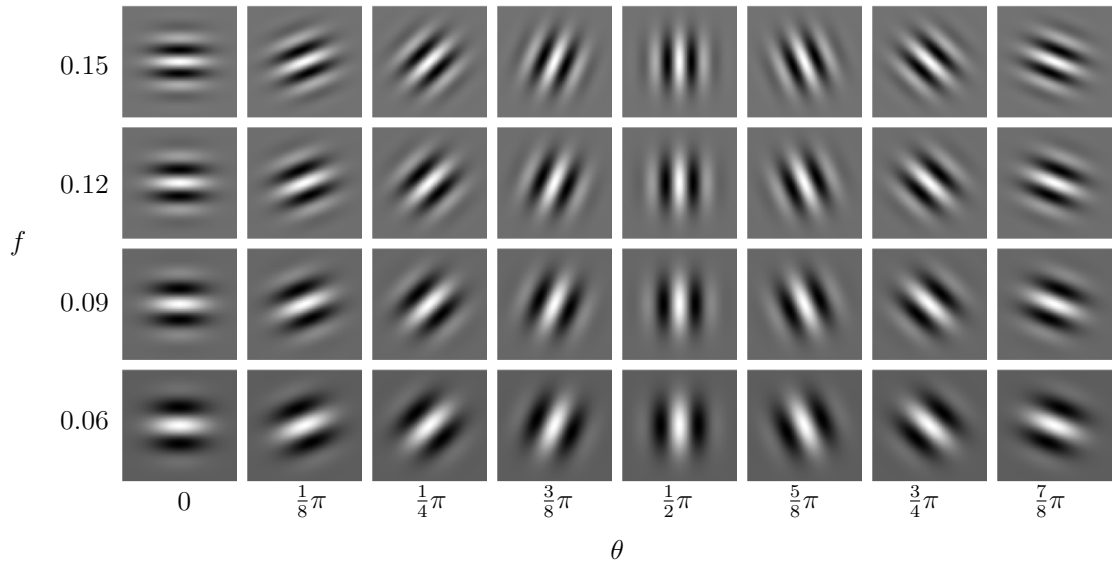
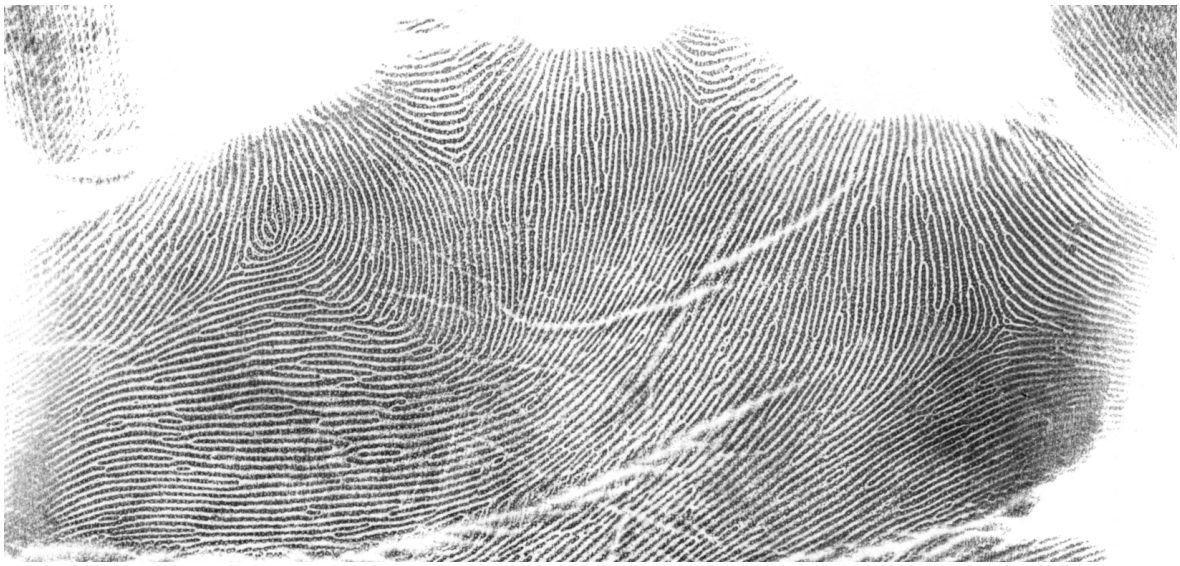
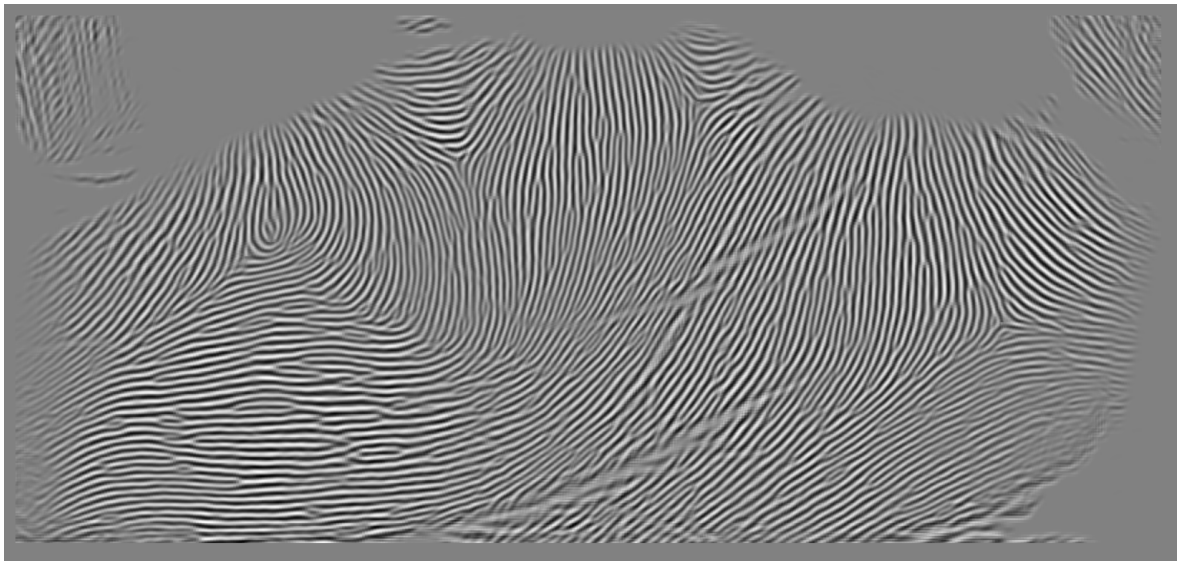


Figure 3.7.: A filter bank of Gabor filters. A graphical representation of Gabor filters for various orientations and densities.

It has been noted that this method introduces artefacts in to the enhanced image [45] and that these artefacts are produced by the mapping of the even-symmetric Gabor filter on to a square grid. To reduce these artefacts *Log Gabor* filters have been proposed [45]. However there are still difficulties associated in matching of the filter to the characteristics of the block, which are yet to be overcome. An alternative potential solution, is to map the even-symmetric Gabor filter on to a circular mask [46] instead of a square block. This reduces the number artefacts, but does not eliminate them. The process of thresholding the image during feature extraction, reduces the number of artefacts further and this will be discussed in section 3.3. An example of contextual filtering using the methods described in this section is illustrated in figure 3.8.



(a) Original palmprint



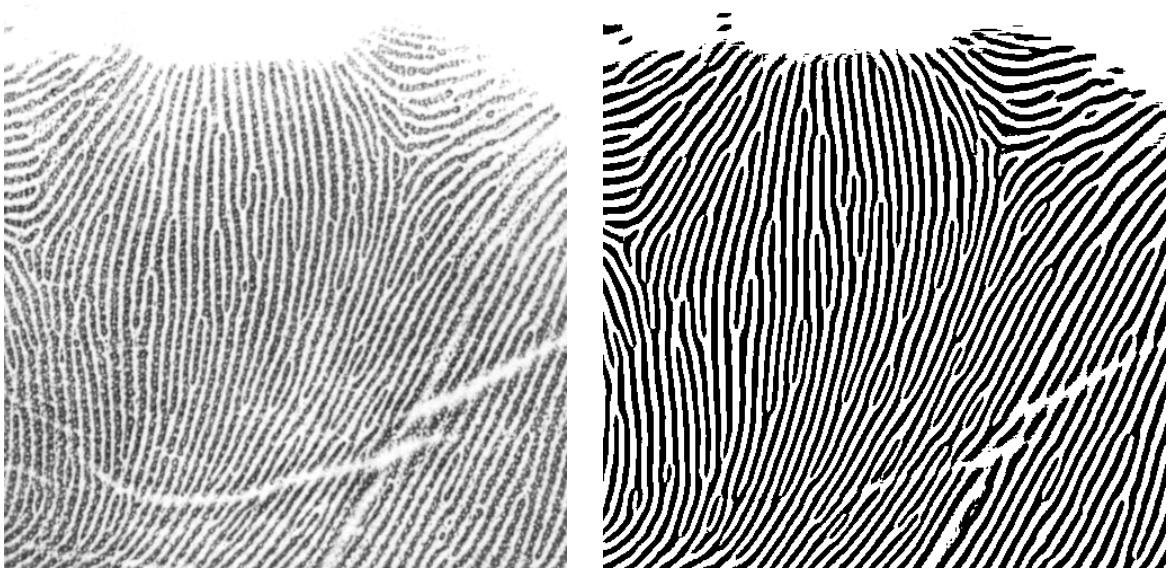
(b) Contextual filtered palmprint

Figure 3.8.: Contextual filtering

3.3. Feature Extraction

The feature extraction methods described in this section refer only to the identification and location of minuatae. Level 1 and Level III features extraction is dealt within in section 3.6. The methods each follow the same general set of steps and rely on a thinned binary image to allow the extraction of the minutiae points, which are then identified and classified using Rutovitz's crossing number [25], which is explained in section 3.3.2.

The enhanced image is thresholded to produce a binary image which represents the peaks and valleys of the friction ridges on the palm or finger print. It is rarely stated in the the literature which threshold calculation method is used. Figure 3.9 illustrates part of an palm and its Gabor enhanced binary counterpart in which the threshold was chosen manually.



(a) Original palmprint (b) Binary, enhanced palmprint

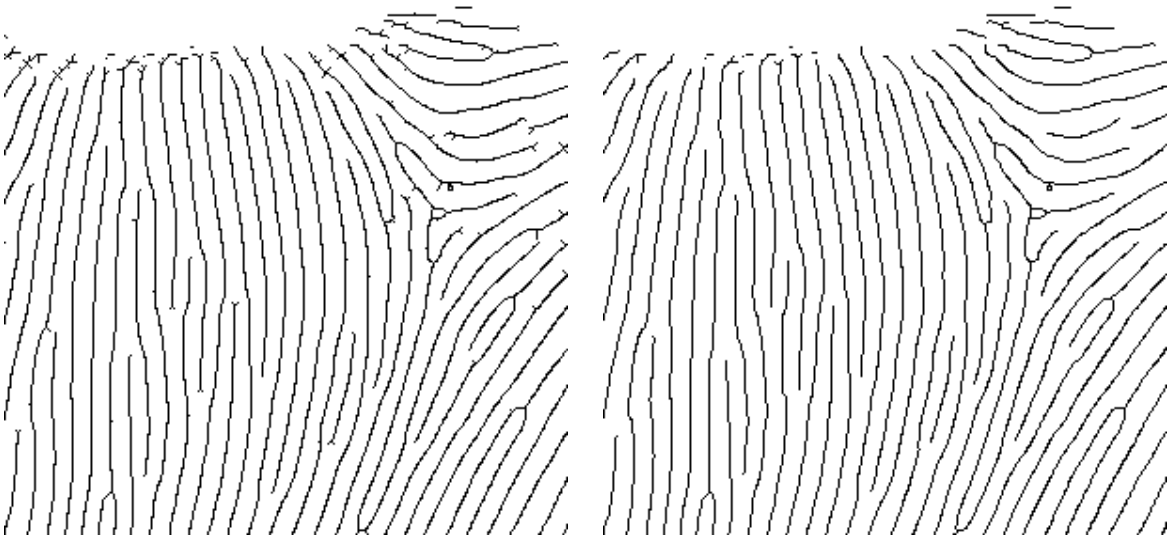
Figure 3.9.: Part of an enhanced binary palmprint

3.3.1. Thinning Binary Images

The *skeletonisation* and *thinning* of a binary image plays an important role in image processing [47]. As a consequence, many techniques exist which thin binary images [48]. Thinning techniques result in a thin skeleton which preserves the end points and central structure of a binary feature.

A large proportion of the APIS [39, 49, 34, 28, 50] use the thinning technique described in [51], known as 'A rotation invariant rule-based thinning algorithm for character recognition'. The same method is also used to thin binary images in AFIS. This is an iterative method which uses a rule based thinning algorithm to prune the boundary of the friction ridges in the binary image. An alternative algorithm [52], has used in APIS [27] due its improved accuracy [53].

Figure 3.10 illustrates the results of both thinning routines when performed on the same enhanced binary palmprint image. It is apparent for the figure, whichever method is used, further processing is required to remove spur atefacts and so produce the correct skeleton.



(a) Thinning by the technique described in [51] (b) Thinning by the technique described in [27]

Figure 3.10.: Thinning an enhanced binary palmprint by different methods, where the black and white pixels denote 1's and 0's in the binary skeleton image, respectively.

The need for further processing is two fold. Firstly, to remove the small spurs which protrude from the skeleton which are present in both methods. Secondly, during the minutiae extraction stage, the angle of a minutiae is calculated and in order to accurately estimate this angle, the skeleton image must be 1 pixel in width (this point is explained further in section 3.3.2). Figure 3.11 (a) illustrates a close up of the thinned image using the technique described in [51]. As can be seen, some of the ridges are wider than 1 pixel; in particular, where direction of the ridges is approximately diagonal.

(a) Skeleton image where some features are wider than 1 pixel

(b) Pre-processed skeleton image where the features are 1 pixel in width

Figure 3.11.: Pre-processing the Skeleton image

The method proposed in [54] deals with the problem of reducing the thickness of the skeleton for palmprints. Similar methods have been proposed for fingerprints [55].

The method described in [54] uses a set of deletion masks for particular patterns of 3 by 3 neighbourhoods. The masks are split into two categories: 1) symmetric and 2) non-symmetric and are illustrated in figure 3.12.

0	0	0	0	1	0	0	1	0	0	0	0
0	**1**	1	0	**1**	1	1	**1**	0	1	**1**	0
0	1	0	0	0	0	0	0	0	0	1	0

(a) Symmetric deleting rules

0	0	1	0	0	0	0	0	1
0	**1**	1	0	**1**	1	0	**1**	1
0	1	0	1	1	0	1	1	0

(b) Non-symmetric deleting rules

0	1	1	0	1	*	*	0	0	1	0	0
1	**1**	0	1	**1**	0	1	**1**	0	1	**1**	0
*	0	0	1	0	0	1	1	1	1	1	*

(c) Non-symmetric deleting rules where * denotes a 0 or a 1.

Figure 3.12.: Deleting rules for thinning skeletons, where the 0's and 1's denote white and black pixels, respectively; the bold 1's indicate the pixel to be deleted (changed to 0)

If the neighbourhoods of the skeleton match the masks shown in figure 3.12 (a) then the corresponding central pixel is set to 0. If the neighbourhoods match any of the masks in figure 3.12 (b) or (c) upon rotation, then the corresponding central pixel is set to 0. The symbol * in figure 3.12 (c) denotes 'don't care' and thus result in a match whatever the value.

3.3.2. Minutiae Extraction and Post Processing

Minutiae extraction is widely performed using the concept of the *crossing number* (CN). The crossing number is defined by Rutovitz [25] as:

P_4	P_3	P_2
P_5	P	P_1
P_6	P_7	P_8

$$CN = \frac{1}{2}\sum_{i=1}^{8} |P_i - P_{i+1}| \tag{3.4}$$

Where P_i correspond to the 8-connected pixels about P, $P_i = \{0, 1\}$ and $P_1 = P_9$.

The crossing number of every pixel which is part of the skeleton is calculated, resulting in the classification for each pixel P as presented in table 3.1.

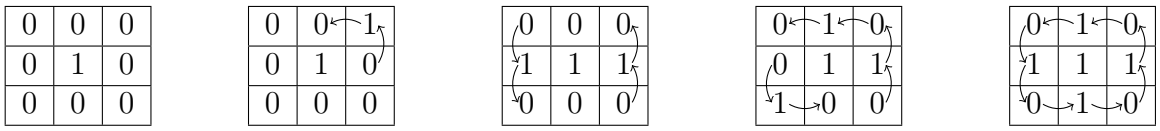
There are 5 categories for the crossing number, each of which is illustrated in figure 3.13. Assuming the skeleton represents friction ridges the classification is as follows. If the crossing number is 0, then the friction ridge is an isolated point; this is illustrated in figure 3.13 (a). If the crossing number is 1 then the pixel lies on the end point of a friction ridge; this is illustrated in figure 3.13 (b). If the crossing number is 2 then the pixel is part of a connected friction ridge; this is illustrated in figure 3.13 (c). If the crossing number is 3 then the pixel lies on a point where the friction ridge bifurcates; this is illustrated in figure 3.13 (d). If the crossing number is 4 then the pixel lies on

CN	Property
0	Isolated point
1	Ending (Minutiae)
2	Connected ridge
3	Bifurcation (Minutiae)
4	Crossing point

Table 3.1.: Rutovitz crossing number

a point where two friction ridges cross; this is illustrated in figure 3.13 (e).

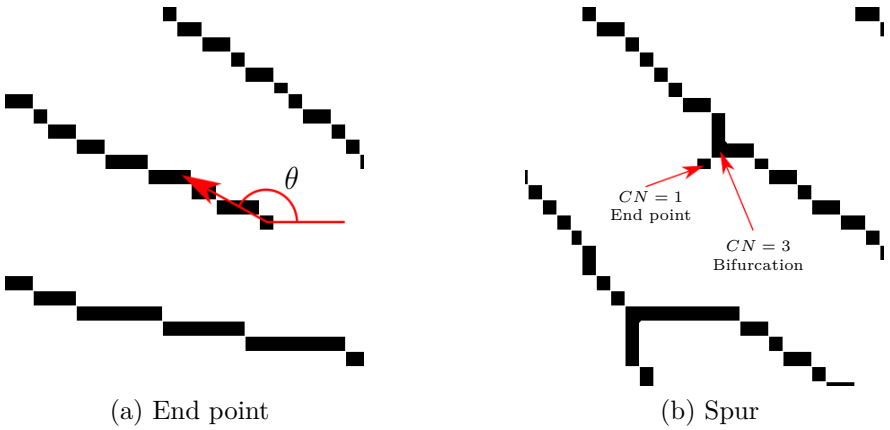
To summarise, any pixel that has a crossing number value of 1 or 3 is defined as a minutiae, either an end point or a bifurcation, respectively.



(a) $CN = 0$ Isolated point (b) $CN = 1$ Ending Minutiae (c) $CN = 2$ Connected ridge (d) $CN = 3$ Bifurcaion minutiae (e) $CN = 4$ Crossing point

Figure 3.13.: The concept of the Rutovitz crossing number

For every minutiae end point ($CN = 1$) in the thinned skeleton, an estimation of its direction is calculated. Given a minutiae ending, the friction ridge is traced away from the minutiae end point along its thinned skeleton [54, 55]. This is illustrated in figure 3.14 by the red arrow which gives the minutiae an angle of θ. The trace moves a set number of pixels (generally 6 pixels) away from the minutiae end point, then calculates the angle of the minutiae by 1) the position of the minutiae end point and 2) the final position of the trace.



(a) End point (b) Spur

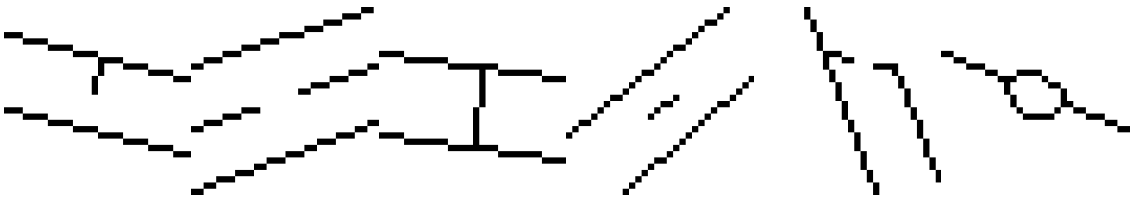
Figure 3.14.: Estimating the direction of the minutiae, by moving 6 pixels along the skeleton

Typically, the position and angle of each minutiae are then used in the matching procedure of an APIS.

3.3.3. Minutiae Filtering

In some circumstances a minutiae end point will lie relatively close to a bifurcation. These features, referred to as *spurs*, occur due to artefacts in the palm or finger print, an example is illustrated by the red arrows in figure 3.14 (b). Spurs introduce minutiae that do not exist in the palmprint image, hence these minutiae are removed.

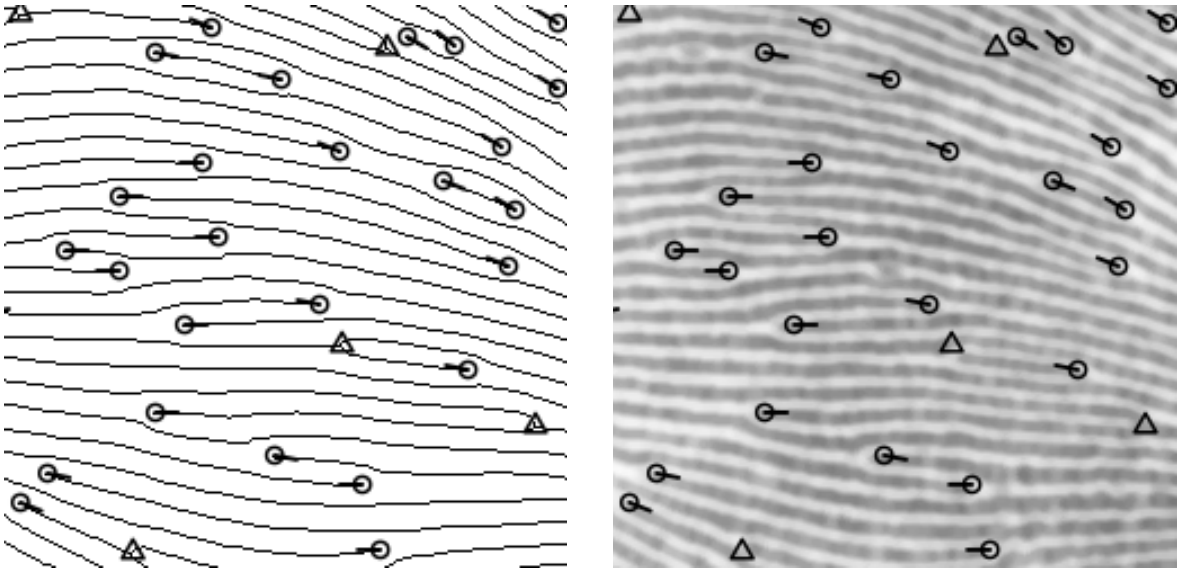
Figure 3.15 illustrates six types of spurious minutiae. Extensions to the original rules used by method [54] (shown in figure 3.12), are used to remove *spurious minutiae*, which are adopted from AFIS.



(a) Spur (b) Break (c) Bridge (d) Island (e) Broken Spur (f) Lake

Figure 3.15.: Examples of spurious minutiae

Examples of the results of automatic minutiae extraction using this method are overlaid upon both the thinned skeleton image and the original palmprint in illustrated in figure 3.16 (a) and (b), respectively.



(a) Minutiae overlaid upon the skeleton (b) Minutiae overlaid upon the palm

Figure 3.16.: Minutiae extraction, where the circle and its line illustrate minutiae end points and their direction and the triangles illustrate bifurcation points.

Each minutiae m is a triple containing positional and directional information, such $m = \{x, y, \theta\}$ where $x, y \in \mathbb{Z}$ and $\theta \in (0, 2\pi]$. The triple which represents each minutiae is now used in the maching stage of APIS.

3.4. Minutiae Matching

Matching minutiae is a problem which has been studied for use in AFIS for many years [6], hence there are many different methods. However, trends in matching philosophies have evolved. As an example, the Hough transform was used to match minutiae [56], by projecting a latent and an exemplars minutiae attributes onto a Hough space and comparing their results. This method works well for *full to full* matching but not for *latent to full*.

When matching latent to exemplar (i.e. *latent to full*) fingerprints, distortions in the friction skin must be considered. The theory of matching *latent to full* fingerprints relies upon the fact that features within a local area preserve their positional invariance [57]. This theory has been exploited for minutiae which exist close to one another, when matching minutiae points.

Since the epidermis on the palm covers more subcutaneous fat [16] than on the epidermis on the fingers, distortions of features contained in palmprints is of greater importance than in fingerprints. Due to this realisation, coupled with some major law enforcement's actively pursuing APIS development [5] at the time, many matching techniques robust to distortions where developed, around 2010. Cutting edge *latent to full* palmprint methods at the time were matching minutiae based upon the triangulation of the position of minutiae [58, 28] and another using MinutiaCode [39].

However, a minutiae matching algorithm [26], called MCC (*Minutiae Cylinder Code*) was proposed in 2010 for fingerprint matching. Each minutiae point is rotated by its associated angle to make it invariant, and local minutiae (size is user specified) are encoded into a cylinder based upon their associated angle and position. Each cylinder from a latent is then matched to each cylinder from an exemplar which gives a local score. A global score is then calculated based upon positional information of each cylinder.

To date there have been over 30 articles and conference papers which attempt to improve upon the matching accuracy in MCC, for example [59, 60, 61].

3.5. Summary

The table 3.2 summarises the APIS methods discussed in this chapter.

APIS	Block Size	θ [1]	Post Filter.	Density Method	Context. Filter.	Features used	Matching Philosophy	Type of Matching	Comments
[50]	16×16	Freq	Yes	Frequency	Gabor	Minutiae Orientation Density	Full to Full/ Latent to Full	Hough, Fusion	*Minutiae removed from areas containing creases*
[58]	Unspecified	Gab[2]	Gabor	Gabor	Gabor	Minutiae	Full to Full	Vertex	*Matching by triangles created by minutiae*
[34]	n/a	n/a	n/a	n/a		Minutiae	Full to Manual	n/a	*Only binary enhancement using Otsu*[3]
[28]	16×16	Slit	Special area[4]	n/a	Filter	Minutiae	Latent to Full	triangulation	*Matching via Radial triangles*
[31]	16×16	Slit		n/a	Filter	Minutiae	n/a	n/a	*Enhancement method only*
[62]	Various	Freq	yes	x-signature	n/a	Orientation Density	Full to Full	pre-aligned	*Study of block sizes*
[23]	Unspecified	Freq	Based on connectivity	n/a	Filter	Ridges	n/a	n/a	*Enhancement technique*
[49]	16×16	Freq	yes	Frequency	Gabor	Minutiae	Full to Full Latent to Full	MinutiaCode	*Local minutiae descriptors*
[39]	16×16	Freq	Growing	Frequency	Gabor	Minutiae	Latent to Full	MinutiaCode	*Region growing*
[17]	64×64	Freq	yes	Frequency	Gabor	Minutiae Orientation Density	Full to Full Latent to Full	Fusion	*Fusion of all three types of feature*
[63]	n/a	n/a	n/a	n/a	n/a	Minutiae SIFT	Latent to Full	Fusion	*No contextual filtering*
[27]	8×8	Grad	yes	x-signature	Gabor	Minutiae	Full to Full	MCC	*A method adopted from AFIS*

Table 3.2.: Summary of the reviewed APIS

[1] Orientation method; i.e. Freq-Frequency, Slit-Slit or Grad-Gradient based method
[2] A Gabor filterbank is used, where θ and density is varied
[3] Otsu threshold is used to create a binary image
[4] Filtering occurs at the centre of Level I structures called "special areas"

3.6. The Potential Use of Level I and III Detail in Current APIS

This section reviews methods which have made use of Level I, Level II and Level III detail in the matching process, and therefore lie outside the standard APIS techniques described in the previous sections. There are very few of these methods and only latterly have they been applied to palms and only then to whole palm matching (i.e not *latent to full*).

A technique using Level III detail was proposed in [64] and uses feature points extracted from the flexion creases of a palmprint. Although there are discriminate features which exist within the flexion creases, these areas of the palm are known to deteriorate with age (see section 2.3.6.3).

3.6.1. Level I detail

The technique described in [50], attempts to use a fusion of the orientation estimate and the friction ridge density of the entire palm (both separately and in combination) with minuate, as the feature set for the matching. Different fusion mechanisms were investigated and included support vector machines and radial basis function neural networks. However, the drawback of this method is that the accuracy quoted was reliant upon a large proportion of the palm being present and therefore, such an approach would not be suitable for latent matching. The method illustrates how other features can be used to compliment minutiae matching.

A different approach [17] for whole palm print matching, separately matches the orientation estimate and friction ridge density and produces a final score using their maximum and individual scores. The results of this approach were promising [62], however, the experiments where performed upon a relatively small set of 40 palmprints and required pre-alignment of the palm print and exemplar.

3.6.2. The Use of Level I Structures in Finger Print Matching

There are no APIS methods in the reviewed literature which use Level I structures within the matching process, however, it is discussed in [50, 18], where it is hypothesised that Level I structures could be useful for indexing or classifying a palm.

Level I structures are widely used in AFIS, where the central point of the structure is often referred to as a singular point [6]. In the reviewed literature Level I structures are used to categorise fingerprints this ease the computational complexity of the matching process [65]. Fingerprints can be classified based upon the frequency and *position* of level I structures. The five main categories of finger print are: Arch, Tented arch, Left loop, Right loop and Whorl [6]. If the size of a latent fingerprint is large enough (i.e. to reveal which category the fingerprint belongs to), then the matching process will only match to fingerprints of that category.

Most of the methods to automatically extract Level I structures in fingerprints can be placed into 2 categories; *template based* or *spatial based*. All of the methods in the reviewed literature operate directly on the orientation field.

The spatial based methods can be further split into two categories; fault line method [66] and the Poincare index [67]. The fault line method detects singular points by quantising the orientation field then segmenting the orientation field into discrete regions [66]. By mapping the boundaries of each of the regions, singular points are

defined by three or more touching regions. This method is illustrated in figure 3.17.

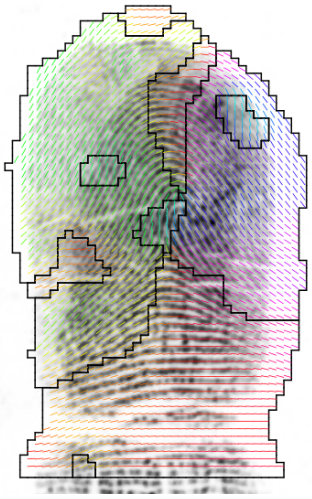


Figure 3.17.: Fault lines of a fingerprint to identify Level I structures

An approach which builds upon the Poincare index [68], uses the consistency of the local friction ridges to detect and classify the singular points in fingerprints. It is specifically designed to detect and classify the category of fingerprint.

Template based methods compare a small local region of the orientation field with a template of the idealised structure. Examples of these methods are proposed in [69, 68]. However [69] concludes that further work is still required if this method is to become invariant to the rotation of the finger print.

An alternative approach proposed in [70] maps the orientation field onto a grayscale colour map and conducts *end of line* analysis to locate the level I structures. However the author concludes that there is a *false-positive* error rate of 17.35%.

3.7. Conclusions

Current APIS used by police forces only exploit a *subset* of minutiae when matching automatically. However, many other features are used by forensic investigators when matching manually. The features used in current APIS (i.e. UK's current APIS, IDENT1) are highlighted in red in figure 3.18.

It has long been recognised [5] that APIS would benefit from the use of additional features and there is a growing trend in recent research to fuse the use of minutiae with other features such as friction ridge orientation and density [17, 62], when matching. However, these new techniques are dominated by *full to full* palm matching applications and are rarely applied to latents. Those techniques which have attempted *latent to full* matching have radically reduced performance [50] when compared to *full to full* matching. There has been no research conducted in to the matching of Level I structures and little research conducted into matching using Level I detail.

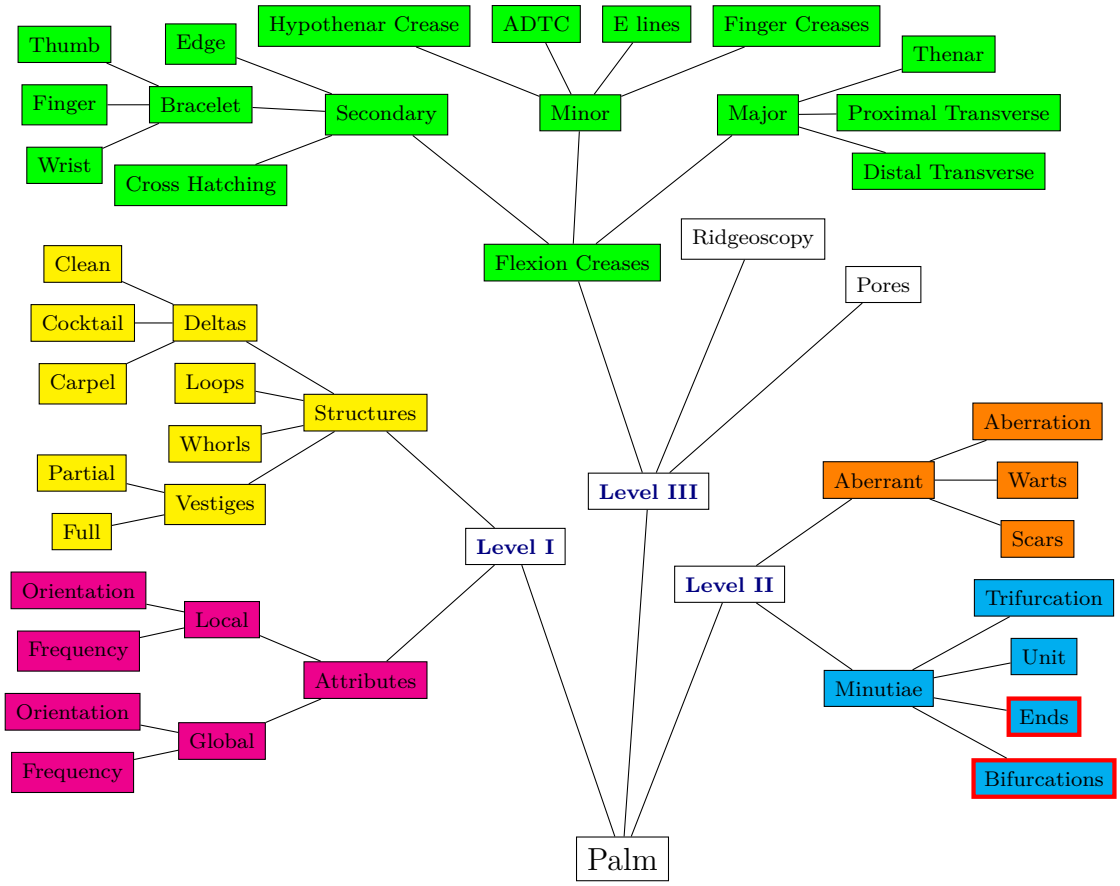


Figure 3.18.: A graphical representation of the detail of a palmprint, where the *only* features used in current APIS are highlighted in red.

The remaining chapters follow the stages described in chapter 3 of the thesis. Chapter 5 describes two new orientation estimation techniques which have been optimised for the extraction of Level I detail. Chapter 7 describes a novel technique which, when applied to the orientation field, locates and classifies the Level I structures within a palmprint. Chapter 7 provides a automatic method to identify and classify areas of Level I detail. The classified Level I detail and structures are then used in a matching process in chapter 8 and applied to a database of palm prints in chapter 8.

Two techniques will be used throughout the rest of thesis for comparison with the work developed here. The first is gradient based method which is adapted from fingerprint based techniques [27] and the other is bespoke method specifically designed for use in palm matching [39].

Chapter 4

Circular Statistics: Review and Framework

Current methods for the derivation and manipulation of orientation estimates are largely ad-hoc [6]. Therefore, owing to the lack of a formal framework, they cannot be readily extended. For example, the definition of the orientation estimate given in [6] (commonly used in AFIS and APIS) does not permit the definition of statistics higher than order two (i.e. variance).

The ad-hoc methods are attempting to solve the problem encapsulated in figure 4.1, called the *opposing orientation problem* [6]. If two opposing vectors are summed in the normal manner, the resulting vector will be zero, (i.e. the vectors cancel). However, what is required is that the summation results in a vector which lies along the red/blue axis. The *opposing orientation problem* is solved by the definition of circular statistics.

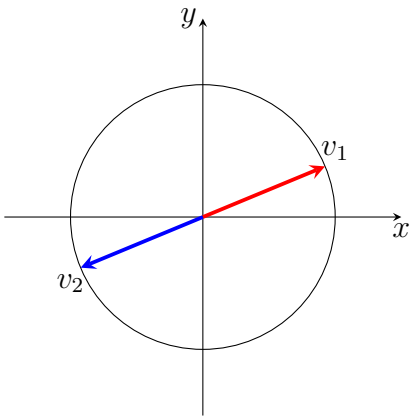


Figure 4.1.: Opposing orientation problem: $v_1 = v_2$

This problem and others, including the derivation of higher order statistics, have been solved within the framework of circular statistics.

The structure of the chapter is as follows. Section 4.1.1 introduces the definition of circular distance and direction, mean and variance. These definitions are extended in section 4.1.1.1 to encompass trigonometric moments for which higher order statistics are derived. Sections 4.1.1.2 and 4.1.1.3 introduce a graphical representation of circular statistics and define axial data, which will be used throughout the thesis. The final section 4.1.2 compares an ad-hoc method of combining the individual orientations, to produce an average orientation estimate, with a method derived from circular statistics. It is shown that these produce equivalent values, however, the circular method can be extended to higher orders and is more computationally efficient.

4.1. Circular Statistics: Methods

Circular statistics are a special type of statistics which deals with the points on the circumference of the unit circle, known as circular data.

Circular data is wrapped around the unit circle and has a period of 2π. This differs from *linear* data as linear data is not periodic. More formally, the support for circular data is the *unit circle*, where as support for linear statistics is the *number line* [71].

The field of circular statistics is well established [72], it is used for much the same purpose as linear statistics. Circular means, medians, variances, measures of dispersion,

etc. are examples of lower order statistics found by circular statistical methods, higher order statistics and tests to fit models to the data are also possible.

The following section describes the necessary circular statistics required for the development of new improved orientation estimates (chapter 5), filtering (chapter 6), feature extraction (chapter 7) and matching (chapter 8).

4.1.1. Circular Statistics

The definitions of circular statistics [72, 71] are now introduced which are calculated upon sets of circular data, which are used in later chapters.

Given two angles θ_1 and θ_2 , d the ***distance between angles*** is defined as:

$$d = \pi - |\pi - |\theta_1 - \theta_2|| \tag{4.1}$$

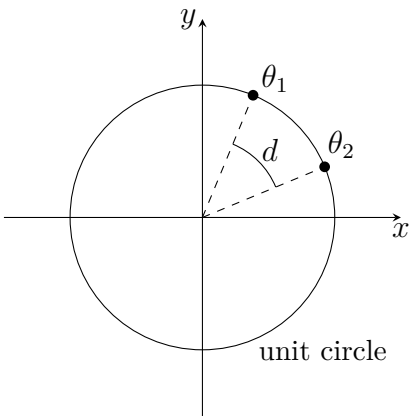


Figure 4.2.: Distance between two angles, θ_1 and θ_2

Suppose we have a set of circular data $\theta_j = \theta_1, \theta_2, \theta_3, ..., \theta_n$ of *cardinality* n (which represent vectors of length 1 from the origin to the unit circle), the ***centre of mass*** in Cartesian coordinates is given by:

$$a = \frac{1}{n}\sum_{j=1}^{n} \cos\theta_j \qquad b = \frac{1}{n}\sum_{j=1}^{n} \sin\theta_j \tag{4.2}$$

Both a and b form the basis from which further circular statistics are calculated. The centre of mass, which can be considered as a point (a, b) is illustrated in figure 4.3.

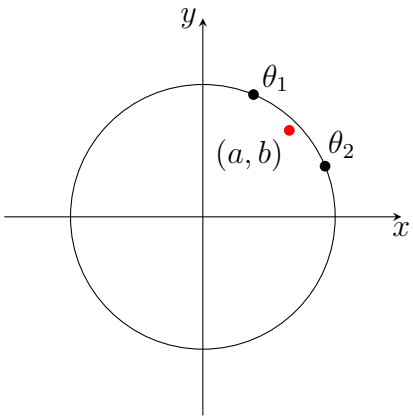


Figure 4.3.: The centre of mass $(a,\ b)$ shown in red, of the two angles θ_1 and θ_2

Then provided that $a > 0$ or $b > 0$, $\overline{R}$ is the ***mean resultant vector***:

$$\overline{R} = \sqrt{a^2 + b^2} \in [0, 1] \tag{4.3}$$

If $a > 0$ or $b > 0$, it follows that $\overline{R} > 0$.

If the data θ_j is clustered closely, then the value of $\overline{R}$ is close to 1, and if θ_j is widely dispersed, the value of $\overline{R}$ is close to 0. $\overline{R}$ is invariant under rotation, and is the most commonly used dispersion statistic.

If $\overline{R} > 0$, then the ***circular mean*** $\overline{\theta}$ is found by:

$$\overline{\theta} = \text{atan2}\,(b, a) \tag{4.4}$$

Then, the ***circular variance*** of θ_j is given by:

$$V = 1 - \overline{R} \in [0, 1] \tag{4.5}$$

Here, V can be thought of as analogous to the variance in linear statistics.

It follows from equation 4.2, when the circular mean $\overline{\theta}$ is subtracted from the set of data θ_j (and when $\bar{R} > 0$):

$$\frac{1}{n}\sum_{j=1}^{n} \cos\left(\theta_j - \overline{\theta}\right) = 0 \qquad \frac{1}{n}\sum_{j=1}^{n} \sin\left(\theta_j - \overline{\theta}\right) = \bar{R} \tag{4.6}$$

This may seem trivial a result, but this property is useful in the calculation for higher order circular statistics, used in the thesis.

4.1.1.1. Trigonometric Moments

Higher order statistics can be calculated using trigonometric moments. The equations 4.2 are key to calculating the circular mean (equation 4.4) and the mean resultant vector (equation 4.3).

It is useful to combine equations 4.2 into what is known as the *first trigonometric moment about the zero direction* $t_{1,0}$ (where the subscript 1 and 0 denotes the first trigonometric moment and the zero denotes zero direction, respectively):

$$t_{1,0} = a + bi \tag{4.7}$$

Then its polar representation is given as:

$$t_{1,0} = \overline{R}e^{i\bar{\theta}} \tag{4.8}$$

This notation is extended to define the p^{th} *trigonometric moment about the zero direction* $t_{p,0}$, for $p = 0 \pm 1 \pm 2 \pm 3, ...$

$$t_{p,0} = \frac{1}{n}\sum_{j=1}^{n}(\cos p\theta_j + i \sin p\theta_j) = a_p + b_p i \tag{4.9}$$

Where:

$$a_p = \frac{1}{n}\sum_{j=1}^{n}\cos p\theta_j \qquad b_p = \frac{1}{n}\sum_{j=1}^{n}\sin p\theta_j \tag{4.10}$$

Then its polar representation is given as:

$$t_{p,0} = \overline{R}_p e^{i\bar{\theta}_p} \tag{4.11}$$

Now $t_{p,0}$ defines the *mean resultant vector* of length:

$$\overline{R}_p = \sqrt{a_p^2 + b_p^2} \in [0, 1] \tag{4.12}$$

with direction:

$$\bar{\theta}_p = \text{atan2}\,(b_p, a_p) \tag{4.13}$$

Now, the trigonometric moment about the circular mean $t_{p,\bar{\theta}}$ (where the subscript p, $\bar{\theta}$ denotes the p^{th} trigonometric moment and the circular mean direction, respectively) is:

$$t_{p,\bar{\theta}} = \frac{1}{n}\sum_{j=1}^{n}\left(\cos p\left(\theta_j - \bar{\theta}\right) + i \sin p\left(\theta_j - \bar{\theta}\right)\right) = \bar{a}_p + \bar{b}_p i \tag{4.14}$$

Where:

$$\bar{a}_p = \frac{1}{n}\sum_{j=1}^{n}\cos p\left(\theta_j - \bar{\theta}\right) \qquad \bar{b}_p = \frac{1}{n}\sum_{j=1}^{n}\sin p\left(\theta_j - \bar{\theta}\right) \tag{4.15}$$

From equation 4.15, when $p = 1$ then $\bar{a}_1 = \bar{R}$ and $\bar{b}_1 = 0$. However, when $p = 2$, $\bar{b}_2$ and $\bar{a}_2$ yield a measure of *skew* and *kurtosis*, respectively. If $\bar{b}_2$ is close to zero, θ_j is symmetric about the mean $\bar{\theta}$, and if $\bar{b}_2$ is negative or positive, θ_j is skewed anticlockwise or clockwise respectively. If $\bar{a}_p = 1$ then all θ_j are equal, (i.e. the highest kurtosis). The *standardised* measures for ***circular skew*** $\hat{s}$ and ***circular kurtosis*** $\hat{k}$ are:

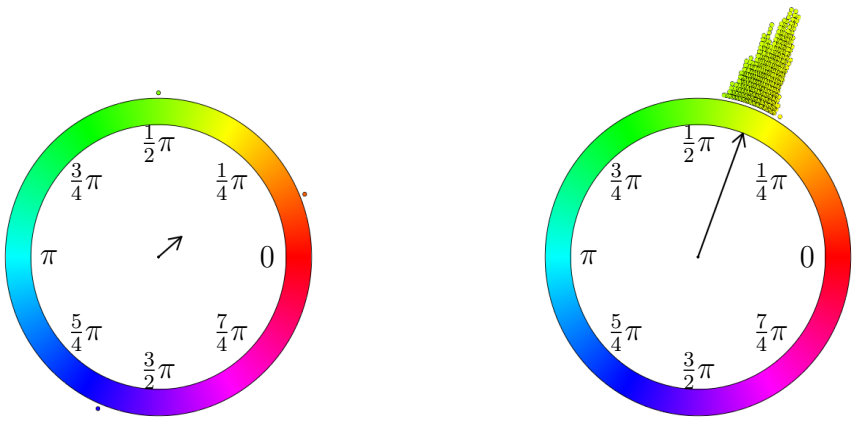
$$\hat{s} = \frac{\bar{b}_2}{\left(1 - \bar{R}\right)^{\frac{3}{2}}} \tag{4.16}$$

$$\hat{k} = \frac{\bar{a}_2 - \bar{R}^4}{\left(1 - \bar{R}\right)^2} \tag{4.17}$$

The interpretation of $\hat{s}$ is similar to $\bar{b}_2$, if $\hat{s}$ is close to zero, θ_j is symmetric about the mean $\bar{\theta}$, and if $\hat{s}$ is negative or positive, θ_j is skewed anticlockwise or clockwise respectively. However, if θ_j is close to the normal distribution $\hat{k}$ will be close to zero, this is because $\bar{a}_2 - \bar{R}^4$ (i.e. when $\bar{a}_2 \simeq 1$ so is $\bar{R}^4 \simeq 1$).

4.1.1.2. Graphical Representation of Circular Statistics

Circular data is normally displayed on what is known as a ***circular histogram***. Figure 4.4 illustrates two sets of data on circular histograms, where a rainbow colour map has been used to display the data. Each particular component from a set of data θ_j is represented by a dot on the outside of the colour map. The length of the arrow indicates the value of $\overline{R}$ and the orientation of the arrow is the circular mean $\overline{\theta}$.



(a) $n = 3$, $\overline{R} = 0.2365$ and $\overline{\theta} = 0.7032$ (b) $n = 1500$, $\overline{R} = 0.9977$ and $\overline{\theta} = 1.2099$

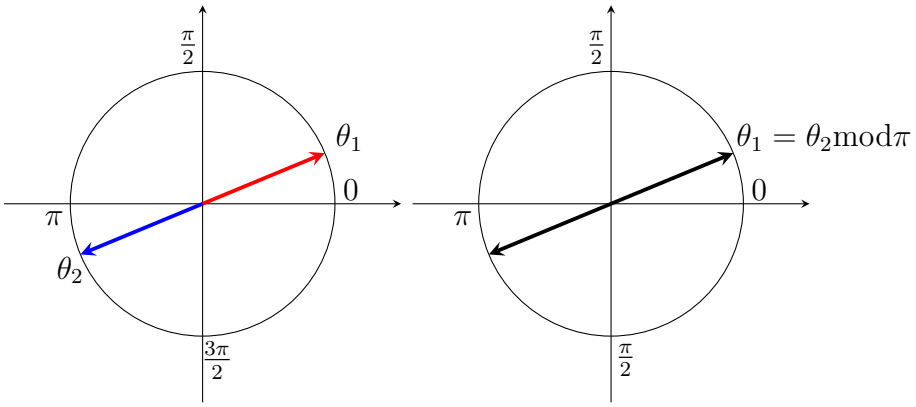
Figure 4.4.: Circular histogram, where each dot on the outside of the rainbow colour map represents the value of an individual element

Figure 4.4 (a) illustrates a set of data of *cardinality* three, where all of the orientations are widely dispersed. Figure 4.4 (b) illustrates a set of *cardinality* 1500, where all of the orientations fit a purposely skewed normal distribution; the skew is -0.9027 (skewed in the anticlockwise direction) and the kurtosis is -0.8153.

4.1.1.3. Axial data

Axial data is well known in circular statistical methods and is the study of circular axes [71]. Since the angle of opposing vectors (see figure 4.5) *share the same axis* (i.e. $\theta = \theta + \pi$). It follows that for any axes, the orientation can be expressed as $\theta = \theta \bmod \pi$. Therefore, axial data spans the range $\theta = [0, \pi)$. A single *axis* can be interpreted as

two opposing vectors because by the definition, $\theta = \theta + \pi$. This is illustrated in figure 4.5 (b).



(a) Circular data: θ_1 and θ_2 (b) Axial Circular data: θ_1 and θ_2 are equivelent

Figure 4.5.: The relationship between circular and axial data

This is the opposing orientation problem, discussed in the introduction to this chapter.

The method for calculating the mean of axial data is simply *doubling* all of the angles in a set, treating it as circular data, calculating the mean, by equation 4.4, then *halving* the result. This process is illustrated in figure 4.6.

Given two angles, d, the ***direction and distance*** between these angles is defined as :

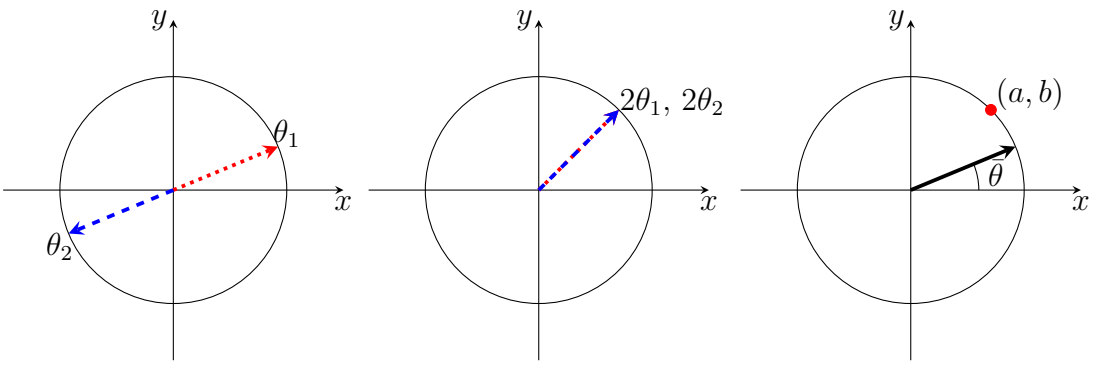
$$\overline{d} = \frac{1}{2}\left(\pi - |\pi - |2\theta_1 - 2\theta_2||\right) \tag{4.18}$$

So, for a set of axial data θ_j, the ***axial circular mean*** direction , $\overline{\theta}$ is found by:

$$\overline{\theta} = \frac{\mathrm{atan2}\left(b, a\right)}{2} \tag{4.19}$$

where

$$\begin{aligned} a &= \frac{1}{n}\sum_{j=1}^{n} \cos 2\theta_j \\ b &= \frac{1}{n}\sum_{j=1}^{n} \sin 2\theta_j \end{aligned} \tag{4.20}$$



(a) θ_1 and θ_2 (b) $2\theta_1$ and $2\theta_2$ (c) Centre of mass of $2\theta_1$ and $2\theta_2$ (red) and final estimate $\bar{\theta}$

Figure 4.6.: Opposing orientation problem: $\theta_i = \left\{\frac{\pi}{8}, \frac{9\pi}{8}\right\}$

All other statistics defined for circular data apply equally to axial data, without the need for further modifications [72].

4.1.2. Comparison of the Fingerprint and Axial Circular Statistics Method for use in Orientation Estimation

The concept of 'doubling the angles' [73] is now widely used in friction ridge analysis, as an attempt to solve the 'opposing orientation problem' (described at the start of chapter 4). The idea is to double the angles of each orientation estimate so that opposing orientation estimates are now in-line with each other. The data will now lie in the range $0 \leq \theta < 4\pi$, this is equivalent, in axial data terms, to wrapping the data twice about the circular axis. The method calculates the average orientation $\overline{\theta}$ from the set of orientation data θ_i (similar to that shown in figure 4.3) using equation 4.21 below.

$$\overline{\theta} = \frac{\pi}{4} + \frac{1}{2}\text{atan2}\left(2G_{xy}, G_{xx} - G_{yy}\right) \tag{4.21}$$

Where:

$$\begin{aligned} G_{xy} &= \sum \cos\theta_i \sin\theta_i \\ G_{xx} &= \sum \cos^2\theta_i \\ G_{yy} &= \sum \sin^2\theta_i \end{aligned}$$

This has led to what is known as the *coherence* [56], which is said to be *measure of dispersion*, denoted as r:

$$r = \frac{\sqrt{\left(G_{xx} - G_{yy}\right)^2 + 4G_{xy}^2}}{G_{xx} + G_{yy}} \tag{4.22}$$

As was shown in section 4.1.1.3, the *opposing orientation problem* is solved formally by the definition of the *circular mean when using axial data*, equation 4.19. Hence the

result of both equations 4.21 and 4.19 are identical, given the same set of data θ_i. In addition, the *mean resultant length* of data θ_i produces identical results to that of the *coherence* of data θ_i.

To compare the computational efficiency of the two techniques a small scale experiment has been performed. For the purpose of this comparison, the methods are referred to as the *Fingerprint method* and the *Axial circular statistics method.* Six experiments were performed, one thousand times for the same set of random data, for each method. Figure 4.7 illustrates the results of the experiment on a log-log scale, where it can be seen that the axial circular statistics method is the more efficient. This may be the result of the additional computation required to calculate the additional third parameter in the finger print method.

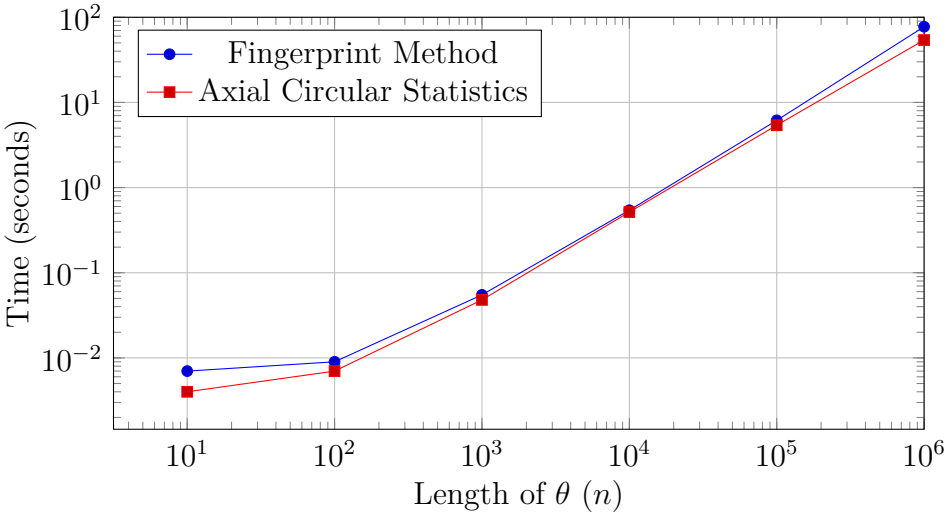


Figure 4.7.: Execution time for mean and dispersion

4.1.3. Conclusions

The chapter has introduced the concept of circular statistics. The use of the appropriate circular statistics has been shown to produce equivalent results to an ad-hoc method for calculation of the orientation, whilst reducing computational complexity. However, unlike ad-hoc approaches, the formal frame work of circular statistics has extensions for higher order statistics.

It will be shown in chapter 7 that higher order statistics can be used in the identification of Level I structures.

Chapter 5

Orientation Estimation

Since the features in both APIS and AFIS are derived from the orientation field, the efficacy and accuracy of reliable orientation estimates is a well studied topic [6]. Section 3 shows that the orientation operators broadly fit into three categories of methodology namely; *frequency* based, *slit* based and *gradient* based. Both frequency and slit base methods result in a discrete set of orientation estimates and thus are inherently less accurate than the continuous estimate produced by the gradient based method. Therefore, this chapter will only be concerned with *gradient based methods*.

These methods estimate the gradient in the x and y directions of the image for each pixel and then use gradients to calculate an orientation estimate for *each pixel* in the image plane.

This chapter investigates two different forms of gradient based operator. Firstly, a 3 by 3 isomorphic kernel as used in [74]. It is shown that the design of previous kernels were not optimal for fiction ridge orientation estimation and a method for the derivation of optimal kernels is provided.

Secondly, a Spherical quadrature filter (SQF's) is introduced which provides a better orientation estimates than can be provided by the optimal 3 by 3 isomorphic kernel.

It will shown that both 3 by 3 kernels and SQF's have scaling factors which affect the accuracy of orientation estimates. Manipulating the scale factor is used to match the filter to the wavelength of features and in so doing, improve the accuracy of the estimate.

However, gradient based methods do suffer loss of accuracy in areas of *small changes in intensity* and at maxima and minima (i.e. *peaks* and *valleys*) [47]. At these points the resulting orientation estimates are unreliable. Areas of friction ridges provide relatively high changes in intensity, except at the maxima and minima. In figure 5.1, an image representing a set of 3 friction ridges and a graph representing the greyscale values (z direction in the graph) as a horizontal slice through the friction ridges is illustrated. The peaks (i.e. maxima) are illustrated by the blue arrows, the valleys (i.e. minima) are represented by the red arrows and the small changes in intensity are illustrated by the green arrows. The high change in intensity corresponds to areas which do not contain arrows, which are areas where friction ridges occur.

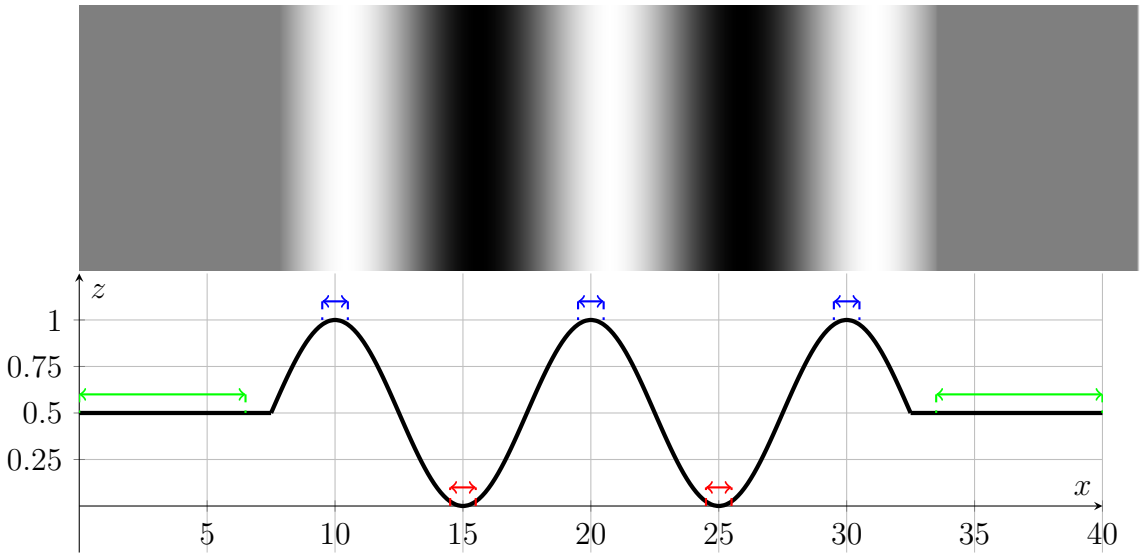


Figure 5.1.: A friction ridge illustrating unreliable estimates : Minima (red), maxima (blue) and low change in intensity (green).

It will be shown later in the chapter that the magnitude of the gradient can be used to identify these unreliable estimates and they can be removed.

The chapter is structured as follows:

- Section 5.1 introduces the test images that shall be used in later sections, to verify the accuracy of gradient based methods.
- Section 5.2 defines a scale factor for 3 by 3 kernel and derives an optimal value for friction ridge orientation and then compares and evaluates the accuracy against of range of selected 3 by 3 kernels.
- Section 5.3 introduces SQF's and their associated scale factor, it then provides a normalisation methodology which allows the comparison of alternative SQFs and finally derives an optimal scale for the chosen SQF.

5.1. Synthetic Test Image

To test the accuracy of gradient based methods, synthetic images are created using a function that approximates the properties of friction ridges. Since the function is defined explicitly, the partial derivatives can also be found explicitly. Thus, the orientation field of the function is known *a priori* and so it can be used to directly compare the accuracy of alternative gradient based methods for each pixel in orientation field.

Analysis similar to the method used in this section has been performed in [75, 76, 74], however, in those cases a Fresnel pattern is used which contains varying wavelengths. In order to better approximate friction ridge flow, this analysis uses a test image containing a *fixed wavelength*, which is then varied over the range of friction ridges wavelengths.

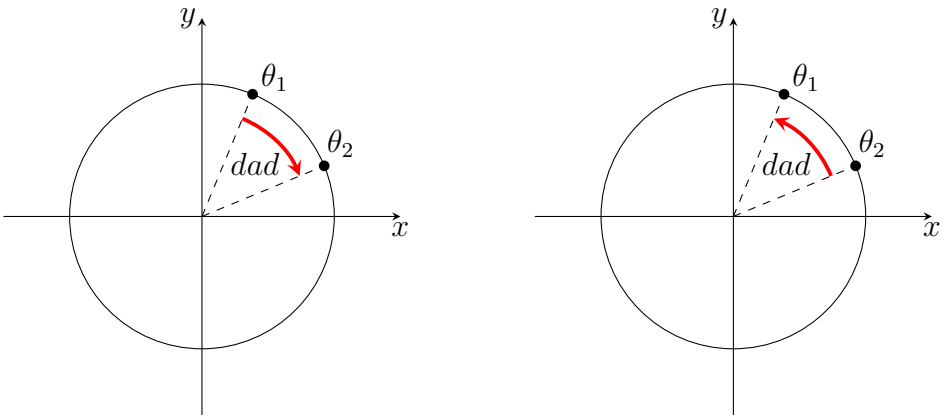
Previous studies have used a measure of the distance between two angles (analogous to equation 4.1) to compare the orientation error of an operator. However, the analysis in this chapter is performed using the ***directional angular distance***, which differs from equation 4.1, only by a measure of direction. This concept is illustrated in figure 5.2,

where it can be seen that positive metrics denote a clockwise direction and negative metrics denote an anticlockwise direction.

The directional angular distance *dad* is defined as,

$$dad\,(\theta_1,\,\theta_2) = \text{atan2}\{\,\sin\,(\theta_1 - \theta_2)\,,\,\cos\,(\theta_1 - \theta_2)\,\} \tag{5.1}$$

It should be noted that the relationship between equation 4.1 and for the angular distance, equation 5.1 is $d = |dad|$.



(a) $dad\,(\theta_1,\,\theta_2) = \frac{\pi}{2}$ (b) $dad\,(\theta_2,\,\theta_1) = -\frac{\pi}{2}$

Figure 5.2.: Directional angular distance between θ_1 and θ_2

5.1.1. Friction Ridge Flow Modelled as Sinusoidal Circular Waves

The test image, $I\,(u, v, \lambda)$, is constructed according to equation 5.2 where u and v $(u,\,v) \in \mathbb{R}^2$ are frequency domain coordinates and λ is said to be the wavelength, measured as the number of pixels per cycle. The range of friction ridge wavelengths are $\lambda = \{7,\,15\}$ [6]

$$I\,(u,\,v,\,\lambda) = \cos\left(\frac{2\pi}{\lambda}\sqrt{u^2 + v^2}\right) \tag{5.2}$$

Figure 5.3 illustrates the relationship between the variables.

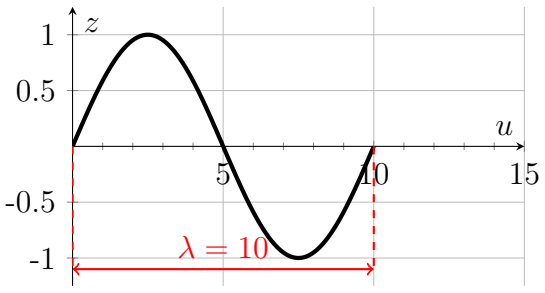


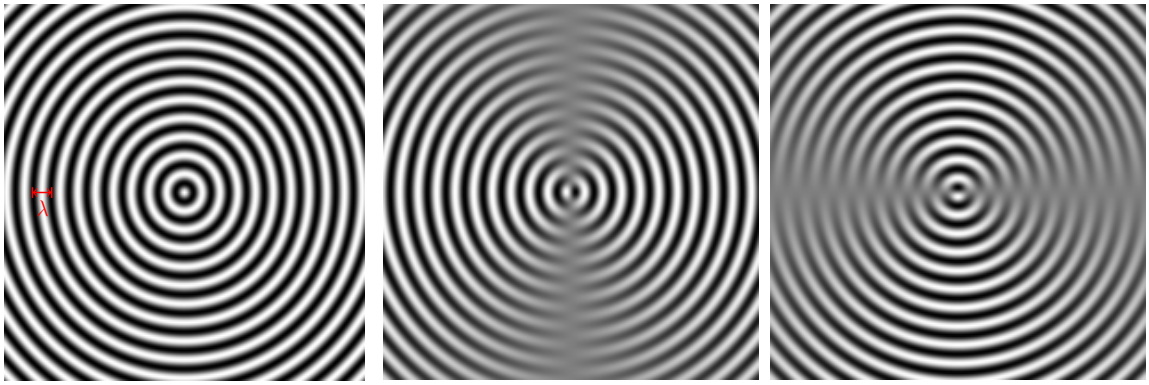
Figure 5.3.: Wavelength and frequency $f = \frac{2\pi}{\lambda}$

The test image generated by equation 5.2 is illustrated in figure 5.4, where it can be seen that the function represents a set of sinusoidal circular waves emanating from the centre of the image.

The partial derivatives of equation 5.2, which define the gradients of $I(u, v, \lambda)$ are given in equations 5.3 and 5.4 illustrated in figure 5.5.

$$\frac{\partial (I(u, v, \lambda))}{\partial u} = \frac{-2\pi u \sin\left(\frac{2\pi}{\lambda}\sqrt{u^2+v^2}\right)}{\lambda\sqrt{u^2+v^2}} \tag{5.3}$$

$$\frac{\partial (I(u, v, \lambda))}{\partial v} = \frac{-2\pi v \sin\left(\frac{2\pi}{\lambda}\sqrt{u^2+v^2}\right)}{\lambda\sqrt{u^2+v^2}} \tag{5.4}$$



(a) $I(u, v, \lambda)$ (b) $\frac{\partial (I(u, v, \lambda))}{\partial u}$ (c) $\frac{\partial (I(u, v, \lambda))}{\partial v}$

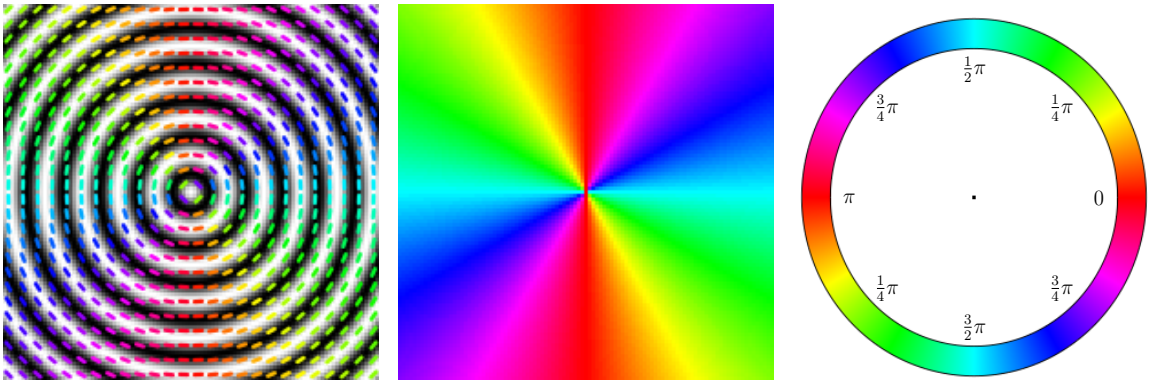
Figure 5.4.: The test image equation 5.2, and its partial derivatives, equations 5.3 and 5.4, respectively.

Thus the axial form of the orientation field, $\theta(u, v, \lambda)$, is given by equation 5.5.

$$\theta(u, v, \lambda) = \left(\frac{\pi}{2} + \frac{1}{2}\text{atan2}\left(\frac{\partial (I(u, v, \lambda)))}{\partial v}, \frac{\partial (I(u, v, \lambda))}{\partial u}\right)\right) \text{mod}\pi \in [0, \pi) \tag{5.5}$$

Where $\theta(u, v, \lambda)$ is now axial data that represents the *real* direction of the flow of the friction ridges.

Figure 5.5 (a) illustrates every fifth orientation estimate as a vector, overlaid upon the synthetic image using the rainbow colour map.



(a) $f(u, v, f)$ (b) $\theta(u, v, f)$

Figure 5.5.: The function $f(u, v, f)$ and its *real* orientation field

5.2. 3 By 3 Kernels

All 3 by 3 gradient operators are isomorphic and are defined by two parameters a and b, as shown in figure 5.6. The individual kernels approximate the gradients in the x and y directions. It should be noted it is not the values of a and b that dictate the accuracy of the orientation field, but rather the ratio of a/b which determines the accuracy of the estimate.

$k_x =$

-b	0	b
-a	0	a
-b	0	b

$k_y =$

b	a	b
0	0	0
-b	-a	-b

Figure 5.6.: 3 by 3 kernels $k(a, b)$

Convolution, denoted by the symbol $\oplus$, is performed on the image, $I(x, y)$, by the individual kernels k_x and k_y, to form the gradient estimates, K_x and K_y, in the x and y directions.

$$\begin{aligned} K_x(x, y) &= I(x, y) \oplus k_x \\ K_y(x, y) &= I(x, y) \oplus k_y \end{aligned} \tag{5.6}$$

$K_x(x, y)$ and $K_y(x, y)$ are then used in equation 5.7 to calculate the orientation estimate, $\theta_{3\times3}(x, y)$, which is an axial estimate of the orientation field.

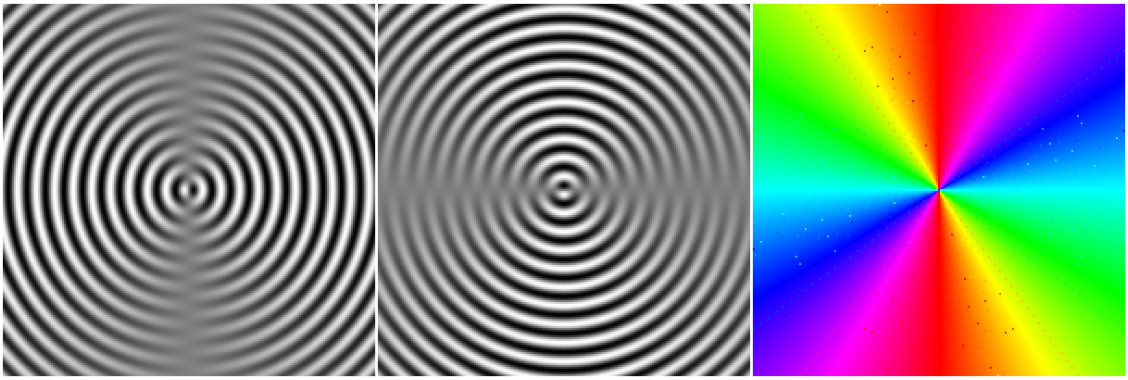
$$\theta_{3\times3}(x, y) = \left(\frac{\pi}{2} + \frac{1}{2}\text{atan2}\left(K_y(x, y), K_x(x, y)\right)\right) \text{mod}\pi \in [0, \pi) \tag{5.7}$$

Figure 5.7 illustrates the gradients $K_x(x, y)$, $K_y(x, y)$ as a result of applying the Sobel kernels to the test image.

5.2.1. Errors of the Orientation field

The Sobel operator [77] is used in this section to highlight the errors which arise from 3 by 3 kernels. The Sobel operator uses the values $a = 2$ and $b = 1$.

Figure 5.7 illustrates the stages for extracting the axial orientation field of the test image using the Sobel operator.



(a) $K_u(x, y)$ (b) $K_u(x, y)$ (c) $\theta_{3\times3}(x, y)$

Figure 5.7.: The stages of applying the Sobel operator to the test image for $\lambda = 10$. (a) and (b) are the vertical and horizontal components of the rates of change and (c) is the resulting orientation field.

The local changes in colour in figure 5.7 (c) illustrate the errors which occur from 3 by 3 operators. These errors appear as pixels which are out of place.

The direction and the size of the errors can be calculated for each pixel by using the circular definition of *directional angular distance* between two angles between $\theta_{3\times3}(x, y)$, and from the actual orientation $\theta(u, v, \lambda)$. To do so we define $dad(\theta(u, v, \lambda), \theta_{3\times3})$ to be the *error of each pixel* of $\theta_{3\times3}$, given θ is the actual orientation field (see section 5.1).

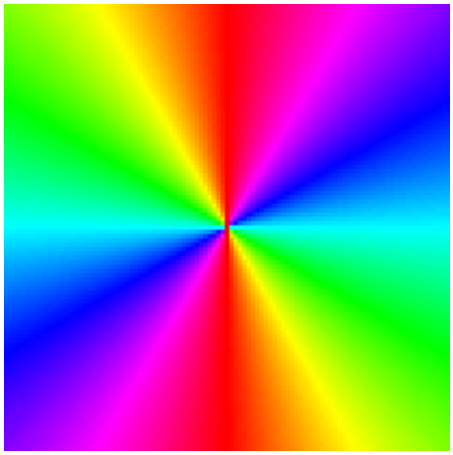
$$\bar{dad}(\theta(u, v, \lambda), \theta_{3\times3}(x, y)) = \frac{1}{2}\text{atan2}\{ \begin{array}{l} sin(2\theta(u, v, \lambda) - 2\theta_{3\times3}(x, y)) \\ \cos(2\theta(u, v, \lambda) - 2\theta_{3\times3}(x, y)) \end{array} \begin{array}{l} , \\ \} \end{array} \tag{5.8}$$

The *average angular distance* $\bar{d}(\theta(u, v, \lambda), \theta_{3\times3}(x, y))$ over the entire orientation field can now be calculated by,

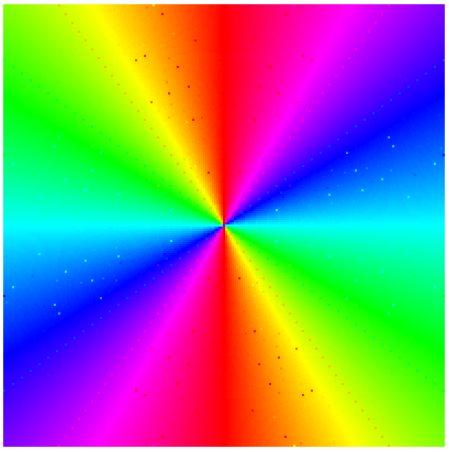
$$\bar{d}(\theta, \theta_{3\times3}) = \frac{1}{n}\sum \left|\bar{dad}(\theta(u, v, \lambda), \theta_{3\times3}(x, y))\right| \tag{5.9}$$

where n is the cardinality of orientation estimates.

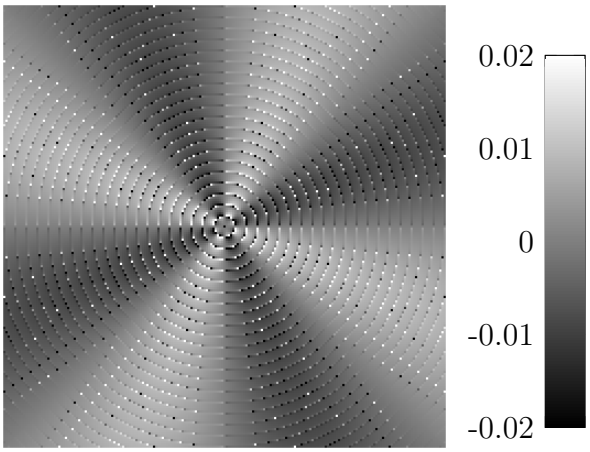
By comparing figure 5.8 (a) with (b) we can see that the Sobel estimated orientation field $\theta_{3\times3}(x, y)$ differs from the actual orientation field $\theta(u, v, \lambda)$.



(a) Actual orientation $\theta(u, v, f)$



(b) Sobel $\theta_{3\times3}(x, y)$



(c) $\bar{dad}(\theta(u, v, \lambda), \theta_{3\times3}(x, y))$

Figure 5.8.: Comparison of the actual orientation and Sobel operator. (c) is the error of each pixel of the orientation field, $\bar{dad}(\theta(u, v, \lambda), \theta_{3\times3}(x, y))$.

Figure 5.8 (c) illustrates $\bar{dad}(\theta(u, v, \lambda), \theta_{3\times3}) \in [-0.02, 0.02]$ using a greyscale colour map; mid grey values represent no errors, lighter pixels represent positive errors and darker pixels represent negative errors. This image illustrates general weakness of gradient based methods and one of the the particular weakness of 3 by 3 kernels. By inspecting figure 5.7 the errors are to be found:

1. At regular intervals (every 5 pixels) emanating radially from the centre of the image, where the minima and maxima of the test image occur.The largest of these errors are approximately $\frac{\pi}{9}$.
2. Where the values of orientation field estimate approach $\pi/8$ and $5\pi/8$, there are negative errors (i.e darker regions) and where the values of orientation field estimate approach $3\pi/8$ and $7\pi/8$ there are positive errors (i.e lighter regions). These errors are due to the miss match between the choice of a/b and λ.

5.2.2. Removing Errors from the Orientation Field Estimate

In order to be able to identify the effects of different kernel ratios (i.e. $\frac{a}{b}$) and quantify their performance, we first must to remove the errors (or unreliable estimates).

By using what is known as the *magnitude of the orientation estimate*, α, (given in

equation 5.10) unreliable orientation estimates can identified and then eliminated.

$$\alpha\left(x,\, y\right) = \sqrt{K_u\left(x,\, y\right)^2 + K_v\left(x,\, y\right)^2} \tag{5.10}$$

The magnitude α can be considered as the strength of the signal at each pixel. If the magnitude is 'high' , then the corresponding orientation estimates are reliable and if the magnitude is 'low', then the corresponding orientation estimates are unreliable.

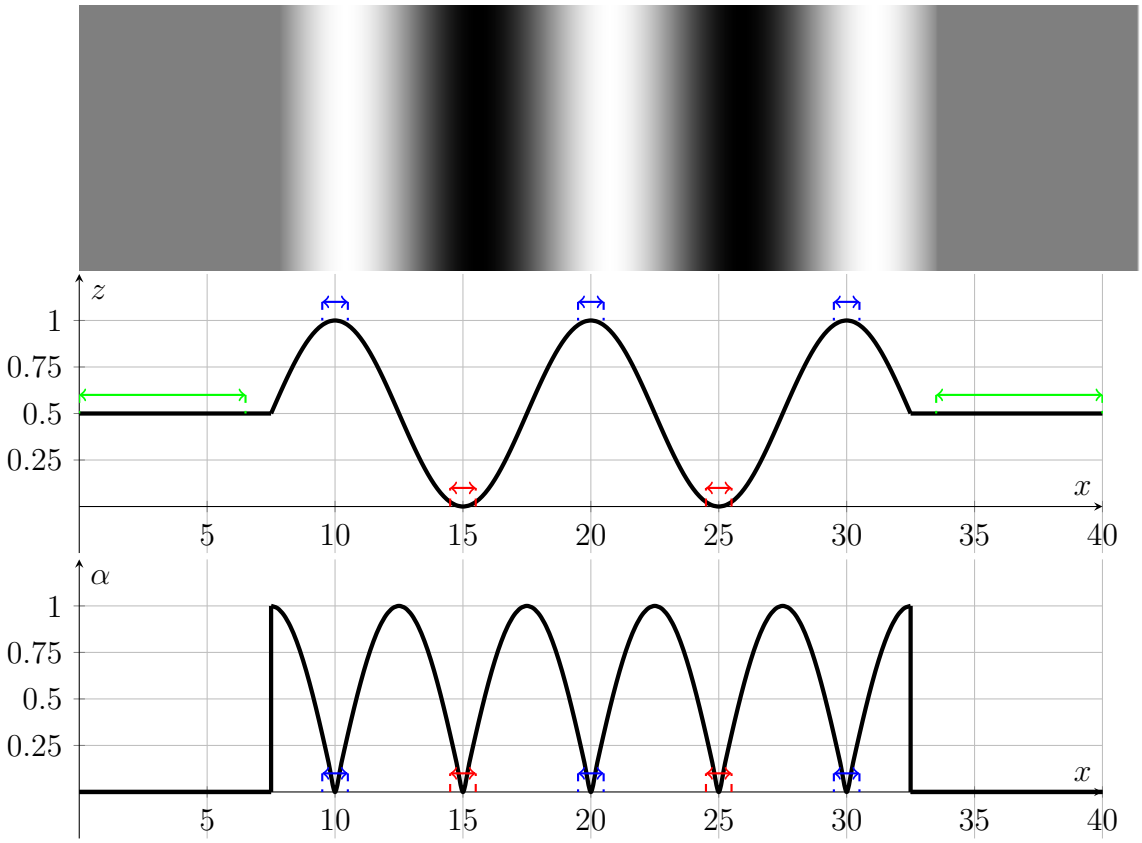


Figure 5.9.: General problems with gradient based methods: Minima (red), maxima (blue) and low change in intensity (green), and where they exist in the magnitude

This is illustrated in one dimension in figure 5.9, where it can be seen that both the maxima (marked blue) and minimas (marked in red) of a sinusoid correspond to the minima of its magnitude.

The same principal is illustrated in two dimensions using the test image in figure 5.10. Figure 5.10 illustrates that the magnitude is 'low' (i.e. the black and white areas) at its minima and maxima and grey elsewhere.

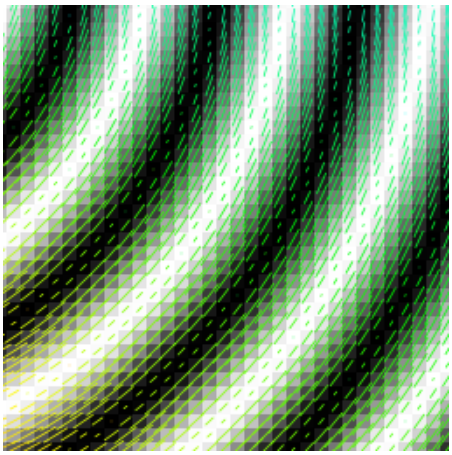


Figure 5.10.: Test image and the orientation represented as segments; where direction and colour represents orientation and length represents magnitude

Given that the values of the magnitude depend upon the absolute values contained in the kernel, in order to be able to isolate unreliable estimates it is necessary to normalise the magnitude. Equation 5.11 provides a normalised estimate , $\overline{\alpha}$, in the range $[0, 1]$.

$$\overline{\alpha}(x,\, y) = \frac{\alpha(x,\, y) - \min(\alpha(x,\, y))}{\max(\alpha(x,\, y)) - \min(\alpha(x,\, y))} \in [0,\, 1] \tag{5.11}$$

Figure 5.11 illustrates a plot of $\bar{dad}$ (the *error per pixel)* against $\overline{\alpha}$ for the Sobel operator. In the ideal case would be a horizontal line at $\bar{dad} = 0$ (i.e. no errors). However, it can be seen in figure 5.11 that there is a spread of $\bar{dad}$ (i.e. the unreliable estimates) and this spread is contained within the interval $\overline{\alpha} \in [0, 0.5]$, as indicated by the red arrow.

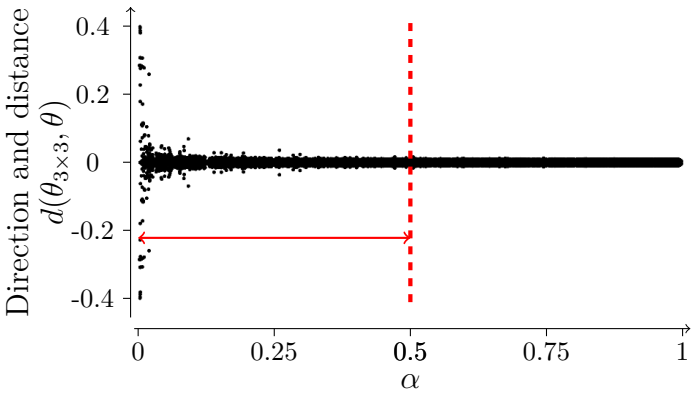
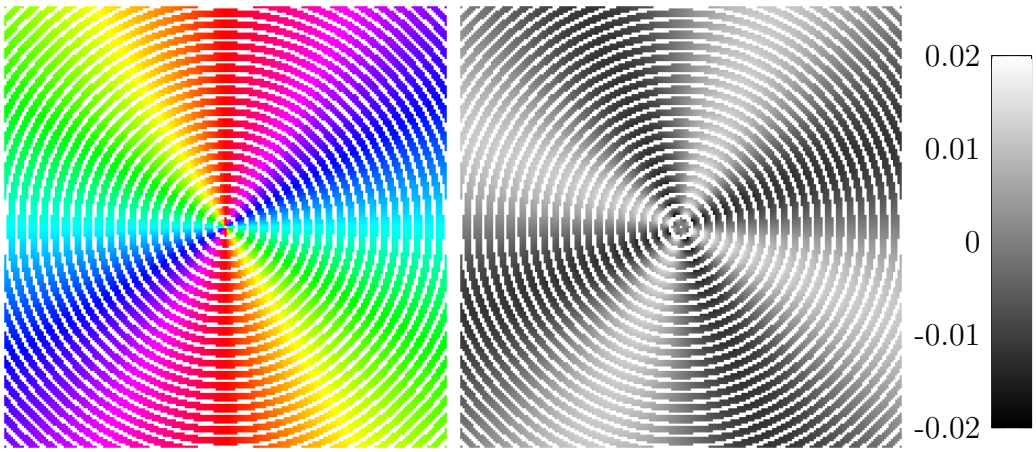


Figure 5.11.: Sobel's orientation field; direction and distance (equation 5.8) against normalised magnitude (equation 5.11)

Therefore, we can improve the average error per pixel ($\bar{dad}$) by removing those orientation estimates with correspondingly low magnitudes, using equation 5.12.

$$\overline{\theta}_{3\times3}(x,\, y) = \begin{cases} \theta_{3\times3}(x,\, y) & \text{if} \quad \overline{\alpha}(x,\, y) > \dfrac{1}{2} \\ NaN & \text{if} \quad \overline{\alpha}(x,\, y) \leq \dfrac{1}{2} \end{cases} \tag{5.12}$$

We choose 0.5 as the threshold in order to remove the largest errors in the test image. This corresponds to approximately 30% of the data. The effect of *low magnitude removal* upon the orientation field estimate for the Sobel operator can be seen in figure 5.12, where the average error per pixel is shown for the Sobel operator with and without *low magnitude removal*.



(a) $\overline{\theta}_{3\times3}(x, y)$ (b) $d(\theta, \overline{\theta}_{3\times3})$

Figure 5.12.: The orientation field $\overline{\theta}_{3\times3}(x, y)$ and its average error $d\bar{a}d$ after low magnitude removal

5.2.3. Optimal ratio for the specific wavelength of features

In this section we derive a relationship between the kernel ratio and the underlying wavelength of the image. The results of this section will show that by optimising this relationship it is possible to produce more accurate orientation estimates. The performance of optimal kernel is then compared with alternative approaches [75, 74, 78], in terms of the average error per pixel. This section assumes that low magnitude removal has been performed on the orientation estimate.

The relationship between the kernel ratio and the image wavelength can be illustrated for a particular wavelength by calculating the average error for a series of test image for kernels with varying b-value. Figure 5.13 illustrates the average error of the orientation field for a test image of $\lambda = 9$ where $a = 1$ (i.e. is fixed) and $b \in [0, 1]$. Figure 5.13 illustrates that such a curve has a single minima which results in the minimum average error for a given λ.

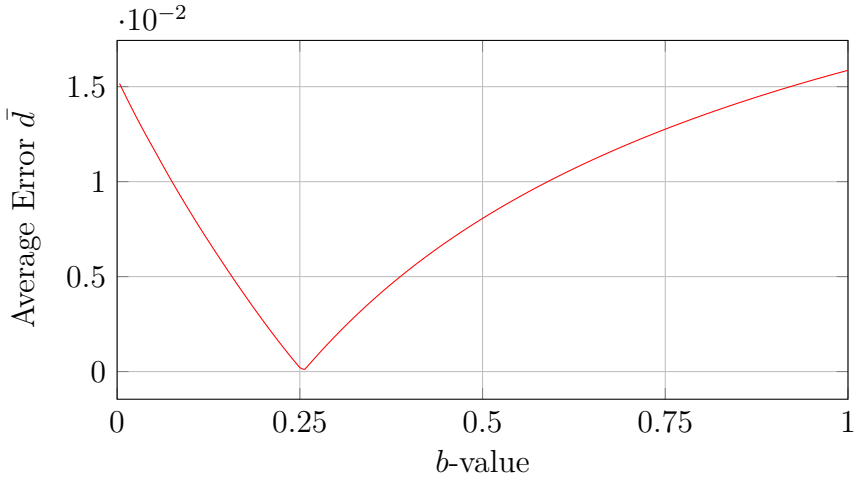
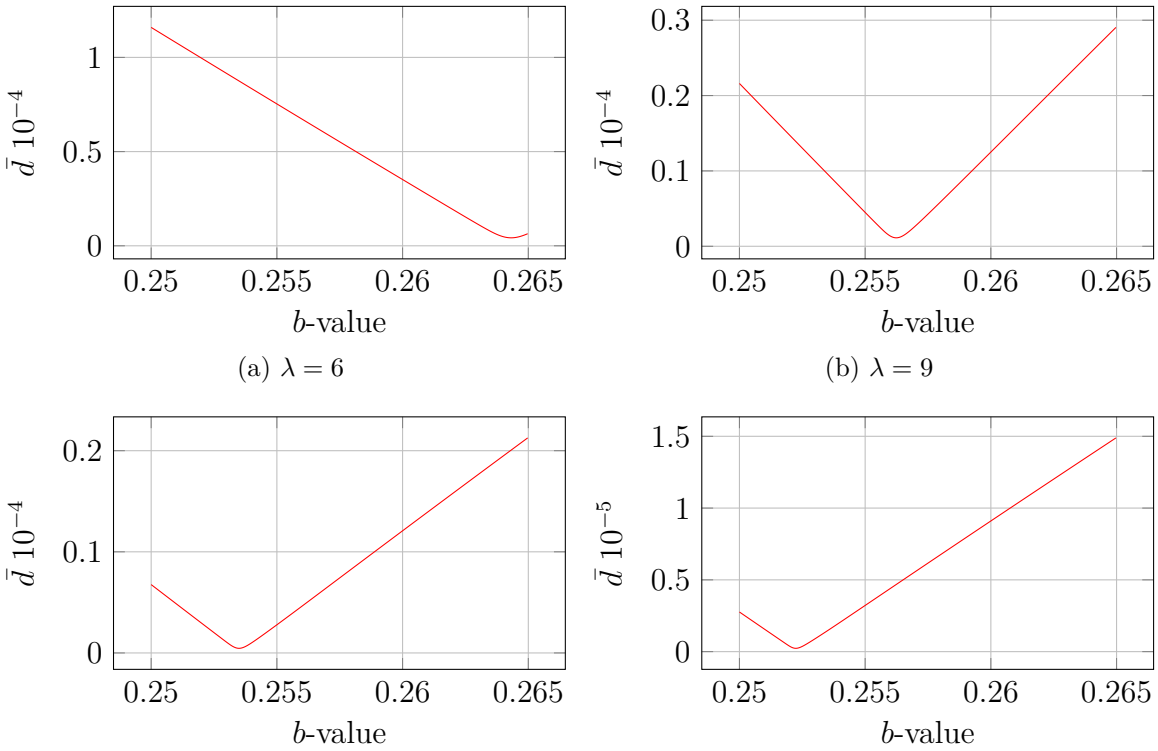


Figure 5.13.: The average error for an orientation field given varying b-value with a fixed wavelength $\lambda = 9$

If the same process is repeated for a range $\lambda \in [6, 15]$, a series of such mimina can be found for each friction ridge wavelength. This series can then be used to define a range of b bounding these minima. Figure 5.14 illustrates the kernel minima for $\lambda = 6, 9, 12, 15$, where it can be seen that optimal b decreases as λ increases.



(a) $\lambda = 6$ (b) $\lambda = 9$

(c) $\lambda = 12$ (d) $\lambda = 15$

Figure 5.14.: Average Error for varying b-value on particular wavelengths λ

Figure 5.15 illustrates the relationship between the optimal b, which minimises the average error, and wavelength, λ, which is described by equation 5.13.

$$\lambda = \frac{0.5143458}{\sqrt{b - 0.2499195}} \tag{5.13}$$

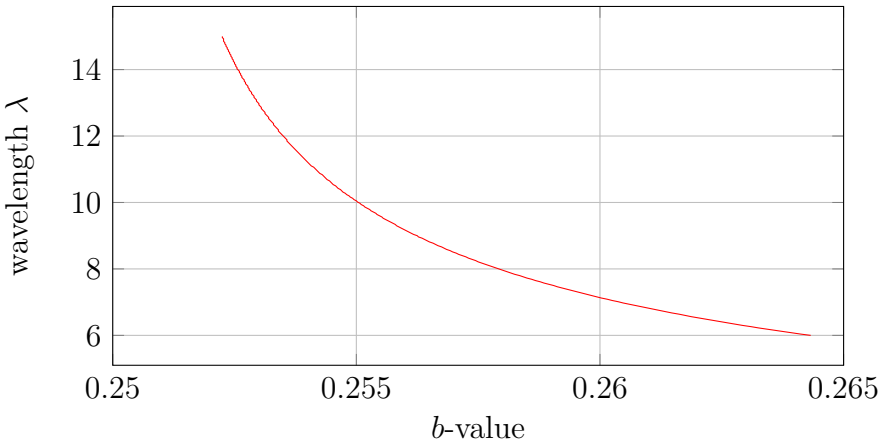


Figure 5.15.: b-value that minimises the error for a given λ

5.2.4. A Comparison of Error with other Isomorphic Kernels

This section compares the optimal filter λ_{opt}, derived using equation 5.9, with the performance of the isomorphic kernels, Sobel, NO[75] , IO [74] and Scharr [78] and an Isotropic kernel [74], each of which have been used to extract the orientation estimates of finger or palm prints. The comparison without low magnitude removal is made using the test image in terms of the *average error,* $\overline{d}$.

-1	0	1
-2	0	2
-1	0	1

1	2	1
0	0	0
-1	-2	-1

(a) Sobel

-17	0	17
-61	0	61
-17	0	17

17	61	17
0	0	0
-17	-61	-17

(b) NO

-1	0	1
-4	0	4
-1	0	1

1	4	1
0	0	0
-1	-4	-1

(c) IO

-3	0	3
-10	0	10
-3	0	3

3	10	3
0	0	0
-3	-10	-3

(d) Scharr

$-\sqrt{2}$	0	$\sqrt{2}$
-1	0	1
$-\sqrt{2}$	0	$\sqrt{2}$

$\sqrt{2}$	1	$\sqrt{2}$
0	0	0
$-\sqrt{2}$	-1	$-\sqrt{2}$

(e) Isotropic

Figure 5.16.: 3 by 3 kernels

Table 5.1 illustrates the average error $\overline{d}$, for λ_{opt} and the 5 kernels, for the values $\lambda = 6, 9, 12, 15$. It can be seen that λ_{opt} ,produces the best orientation estimates for the given values of λ. The λ_{opt} kernel is 4.51, 2.7, 2.47 and 1.16 times more accurate than the next most accurate method over the range of wavelengths 6, 9, 12 and 15 respectively.

	λ			
Operator	6	9	12	15
Sobel	0.016379	0.0077484	0.0044255	0.0038139
NO	0.0014056	0.00098158	0.000591155	0.00054292
IO	0.0011599	0.00022948	0.000068753	0.000028786
Scharr	0.0031644	0.001811	0.0010494	0.00093358
I	0.025775	0.011753	0.00677	0.0056006
λ_{opt}	**0.00025709**	**0.00008515**	**0.000027811**	**0.000024711**

Table 5.1.: Average error of 3 by 3 kernel at specific λ

Figure 5.17 illustrates how the error of each filter varies with angle. The ideal error response would be a horizontal line at $\overline{d} = 0$. It can seen that λ_{opt} best approximates the ideal response (illustrated by the black line).

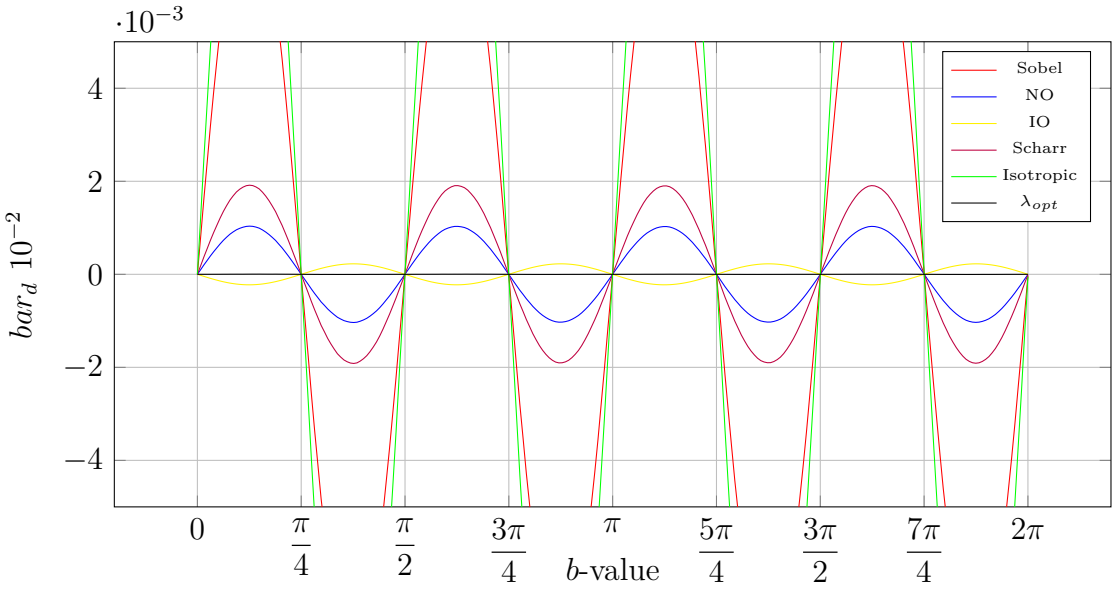


Figure 5.17.: Average error $\overline{d}$ for each kernel at given orientations

5.2.5. Summary

An expression has been derived which describes the optimal relation between the kernel ratio of isomorphic kernels and a single wavelength within the image. It has been shown that the optimal kernel range for friction ridges lies within $b \in [0.2525, 0.2650]$ and that the average error of such filters orientation estimate is better than kernels previously applied to orientation field estimation.

A method for the removal of unreliable estimates, based upon the magnitude of the orientation estimate, has been presented and has been shown to further reduce the average error of such kernels.

The results of this section will be used in section 5.4, to investigate the effect of the use of λ_{opt} and low value magnitude removal in current APIS.

5.3. Spherical Quadrature Filters

Spherical quadrature filters (SQF's) have previously been used as a part of a method for orientation and curvature estimation [79], edge and corner detection [80] and optical flow. Although other approaches can be applied to similar applications, for example convolution with kernels [74], Radon transform [81] etc, SQF's can offer a more robust and accurate approach to the problem of scale space in the image plane [82].

Motivated by obtaining a robust and accurate orientation estimate, this section investigates the use of spherical quadrature filters as orientation estimators for palmprints. SQF's are a set of 2 (or more) isotropic filters that lie in quadrature, in the case of only two filters, the filters lie $\frac{\pi}{2}$ out of phase. This section will investigate the family of SQFs consisting of two isotropic filters. To the author's knowledge, no study exists which compares the performance of SQF's as an orientation operator.

Generally SQF's are constructed using the u and v coordinates in the frequency domain, such that $(u, v) \in \mathbb{R}^2$. A general form for a pair of SQF's is given by:

$$SQF(u, v, \sigma) = \begin{bmatrix} u \\ v \end{bmatrix} f(u, v, \sigma) \tag{5.14}$$

Where σ controls the scale of the filters and $f(u, v, \sigma)$ is some given of distribution.

Larger values of σ produce filters which are more immune to the effects of noise, but their orientation estimates are less localised. Conversely, as σ decreases the filter is less immune to noise but its orientation estimates are more localised [82].

The following sub-sections are structured as follows. Section 5.3.1 introduces the SQFs under investigation. Section 5.3.2 provides a method of normalisation which allows the comparison of their performance. Section 5.4.1 evaluates their use as orientation operators and provides examples of their application to palmprints.

5.3.1. A Family of Spherical Quadrature Filters

In this section a brief rationale is given for the particular SQF's which are used in the comparison, followed by their definitions.

- The properties of the *Gaussian filter* make it a suitable choice for orientation extraction [83], since the frequency response of a Gaussian is itself a Gaussian [76], since invariance to rotation is a desired result for palmprint matching.
- The *monogenic signal* [84] is a low level image processing technique based upon the Riesz transform, in which the Poisson distribution is used to construct the SQF. Further work in this field has led to the *signal multivector* [79] and *conformal embedding* [85].
- An edge detection technique known as Phase Congruency [82] uses a scale unifying approach, that builds upon multi-scale wavelet analysis, using Log-Gabor filters. This method was designed to detect symmetry and anti symmetry in images [80].
- The derivatives and the Laplacian of Gaussian and Poisson SQFs form the basis of a large number of low level image processing applications and therefore, they will also be included here.

We define the following SQF's by constructing them in the frequency domain, such that $(u,\, v) \in \mathbb{R}^2$.

Gaussian Filter:

$$G\left(u,\, v,\, \sigma\right) = \left[\begin{array}{c} u \\ v \end{array}\right] e^{-\left(\frac{u^2+v^2}{\sigma^2}\right)} \tag{5.15}$$

Poisson Filter:

$$P\left(u,\, v,\, \sigma\right) = \left[\begin{array}{c} u \\ v \end{array}\right] \frac{1}{2\pi\left(u^2+v^2+\sigma^2\right)^{\frac{1}{2}}} \tag{5.16}$$

Derivative Gaussian Filter:

$$\partial G\left(u,\, v,\, \sigma\right) = \left[\begin{array}{c} u \\ v \end{array}\right] \frac{2e^{-\left(\frac{\sqrt{u^2+v^2}}{\sigma^2}\right)}}{\sigma^2} \tag{5.17}$$

Derivative Poisson Filter:

$$\partial P\left(u,\, v,\, \sigma\right) = \left[\begin{array}{c} u \\ v \end{array}\right] \frac{3}{2\pi\left(u^2+v^2+\sigma^2\right)^{\frac{3}{2}}} \tag{5.18}$$

Laplacian of Gaussian Filter[1]:

$$\Delta G\left(u,\, v,\, \sigma\right) = \left[\begin{array}{c} u \\ v \end{array}\right] \frac{\left(1-\frac{u^2+v^2}{2\sigma^2}\right) e^{-\left(\frac{u^2+v^2}{2\sigma^2}\right)} - e^{-\left(\frac{u^2+v^2}{\sigma^2}\right)}}{2\pi\sigma^6} \tag{5.19}$$

Laplacian of Poisson filter[1]:

The Laplacian of Poisson filter cannot be expressed in the same form as the earlier definitions (as a product of the coordinates u and v), therefore each filter must be defined separately:

$$\Delta P\left(u,\, v,\, \sigma\right)_u = \frac{-30u}{\pi\left(u^2+v^2+\sigma^2\right)^{\frac{7}{2}}} + \frac{52.5uv^2}{\pi\left(u^2+v^2+\sigma^2\right)^{\frac{9}{2}}} + \frac{52.5u^3}{\pi\left(u^2+v^2+\sigma^2\right)^{\frac{9}{2}}} \tag{5.20}$$

$$\Delta P\left(u,\, v,\, \sigma\right)_v = \frac{30v}{\pi\left(u^2+v^2+\sigma^2\right)^{\frac{7}{2}}} - \frac{52.5u^2v}{\pi\left(u^2+v^2+\sigma^2\right)^{\frac{9}{2}}} - \frac{52.5v^3}{\pi\left(u^2+v^2+\sigma^2\right)^{\frac{9}{2}}} \tag{5.21}$$

Log Gabor Filter:

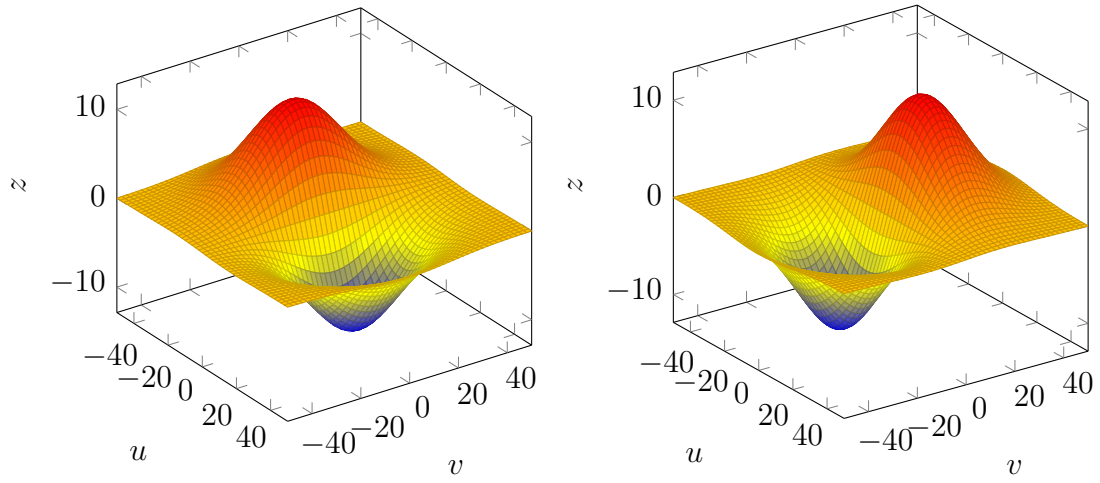
$$LG\left(u,\, v,\, \sigma\right) = -\left[\begin{array}{c} u \\ v \end{array}\right] e^{-\left(\log\left(\frac{\sqrt{u^2+v^2}}{\sigma^2}\right)^2\right)} \tag{5.22}$$

[1]The Laplacian of each filter is an approximation of the difference of the filter.

Derivative of Log Gabor Filter:

$$\partial LG\left(u,\, v,\, \sigma\right) = -\begin{bmatrix} u \\ v \end{bmatrix} \frac{e^{-\left(\log\left(\frac{\sqrt{u^2+v^2}}{\sigma^2}\right)^2\right)} \log\left(\frac{\sqrt{u^2+v^2}}{\sigma^2}\right)^2}{u^2+v^2} \tag{5.23}$$

Therefore for each of the SQF's above, we have two filters that lie in quadrature and a particular value of σ which controls their scale. As an example figure 5.18 illustrates the pair of derivative of Gaussian SQF's (with a $\sigma = 25$) using a heat map to show the values of the filter. The maximum value of the filter is approximately $z = 10$.



(a) $\partial G\left(u,\, v,\, \sigma\right)_u$ (b) $\partial G\left(u,\, v,\, \sigma\right)_v$

Figure 5.18.: Derivative of Gaussian filters constructed in the frequency domain

However, as can be seen in figure 5.19, the σ's in each SQF differ from each other by a scalar factor and therefore, the SQFs cannot be directly compared as they stand. Therefore, to be able to compare the orientation estimates generated by the individual SQFs, their σ's and maximum magnitudes must be first normalised.

5.3.2. Normalising SQF's

The maximum magnitudes of each SQF (equations 5.15 to 5.23) are also not the same. In order to normalise the magnitude of the filters, we first divide by the its maximum value, so as to limit the magnitude between $[-1, 1]$, as illustrated in figure 5.19.

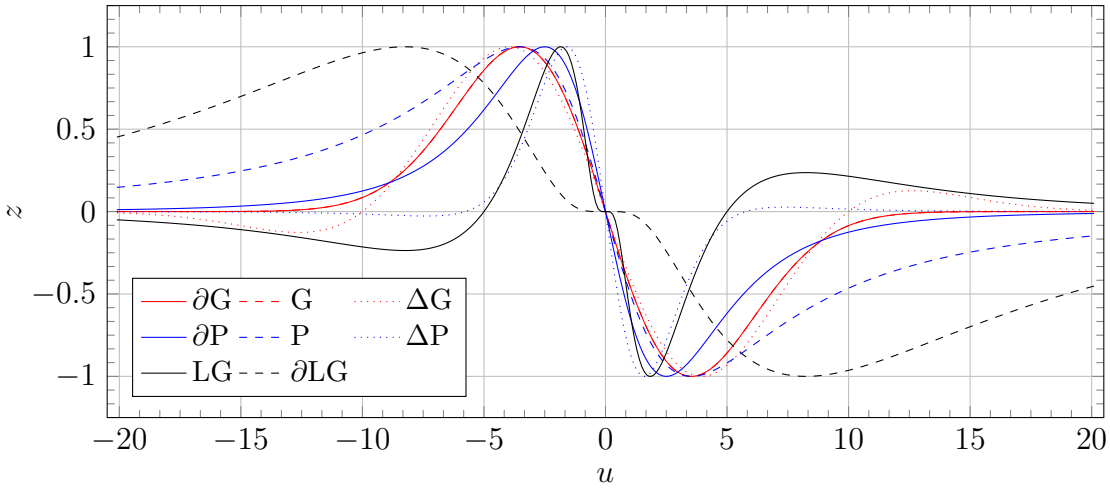


Figure 5.19.: Cross section of the u direction for each SQF, where $v = 0$ from the same σ

Figure 5.19 also illustrates that the maximum magnitudes of each filter occur at different values of u. The position of the maximum magnitudes is controlled by σ. By calculating the derivative of each SQF, we can determine the position of each maximum magnitude in terms of a *normalised* σ, which we will denote $\bar{\sigma}$. The detailed calculations required to normalise σ, for the Laplacian of Gaussian and Poisson filters and so co-locate the maximum magnitudes, are to be found in the appendices.

By replacing σ in the definitions above with the corresponding $\overline{\sigma}$ defined below, we are able to co-locate the position of maximum magnitudes of each SQF, as shown in figure 5.20.

Gaussian normalisation:

$$\bar{\sigma} = \sqrt{2\sigma^2} \tag{5.24}$$

Poisson normalisation:

$$\bar{\sigma} = \sqrt{2\sigma^2} \tag{5.25}$$

Derivative Gaussian normalisation:

$$\bar{\sigma} = \sqrt{2\sigma^2} \tag{5.26}$$

Derivative Poisson normalisation:

$$\bar{\sigma} = 4\sqrt{2\sigma^2} \tag{5.27}$$

Laplacian of Gaussian normalisation:

$$\bar{\sigma} = \frac{\sqrt{\sigma\left(\sqrt{33}+7\right)}}{2^{\frac{3}{2}}} \tag{5.28}$$

Laplacian of Poisson normalisation:

$$\bar{\sigma} = \frac{\sqrt{\sigma\left(\sqrt{1393}+41\right)}}{2^{\frac{3}{2}}} \tag{5.29}$$

Log Gabor normalisation:

$$\bar{\sigma} = \frac{\sigma}{\exp\left\{\frac{1}{2}\right\}} \tag{5.30}$$

Derivative of Log Gabor normalisation:

$$\bar{\sigma} = \sigma \exp\{1\} \tag{5.31}$$

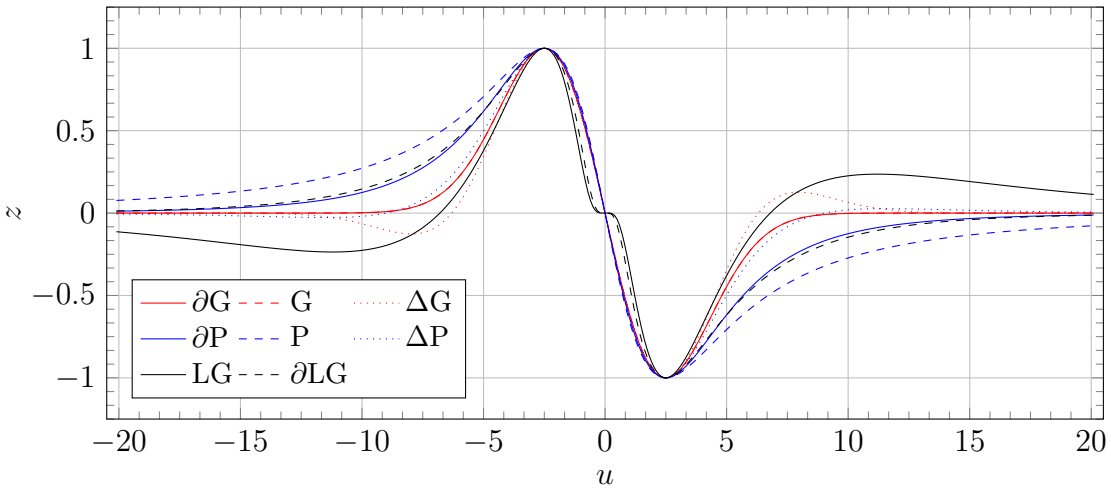


Figure 5.20.: Cross section of the normalised SQF's $\bar{\sigma} = 3$ where $v = 0$

This form of normalisation retains the morphological properties of the original filter, so allowing comparison of the SQFs. Figure 5.21 illustrates each of the normalised filters in two-dimensional using a grey scale colour map.

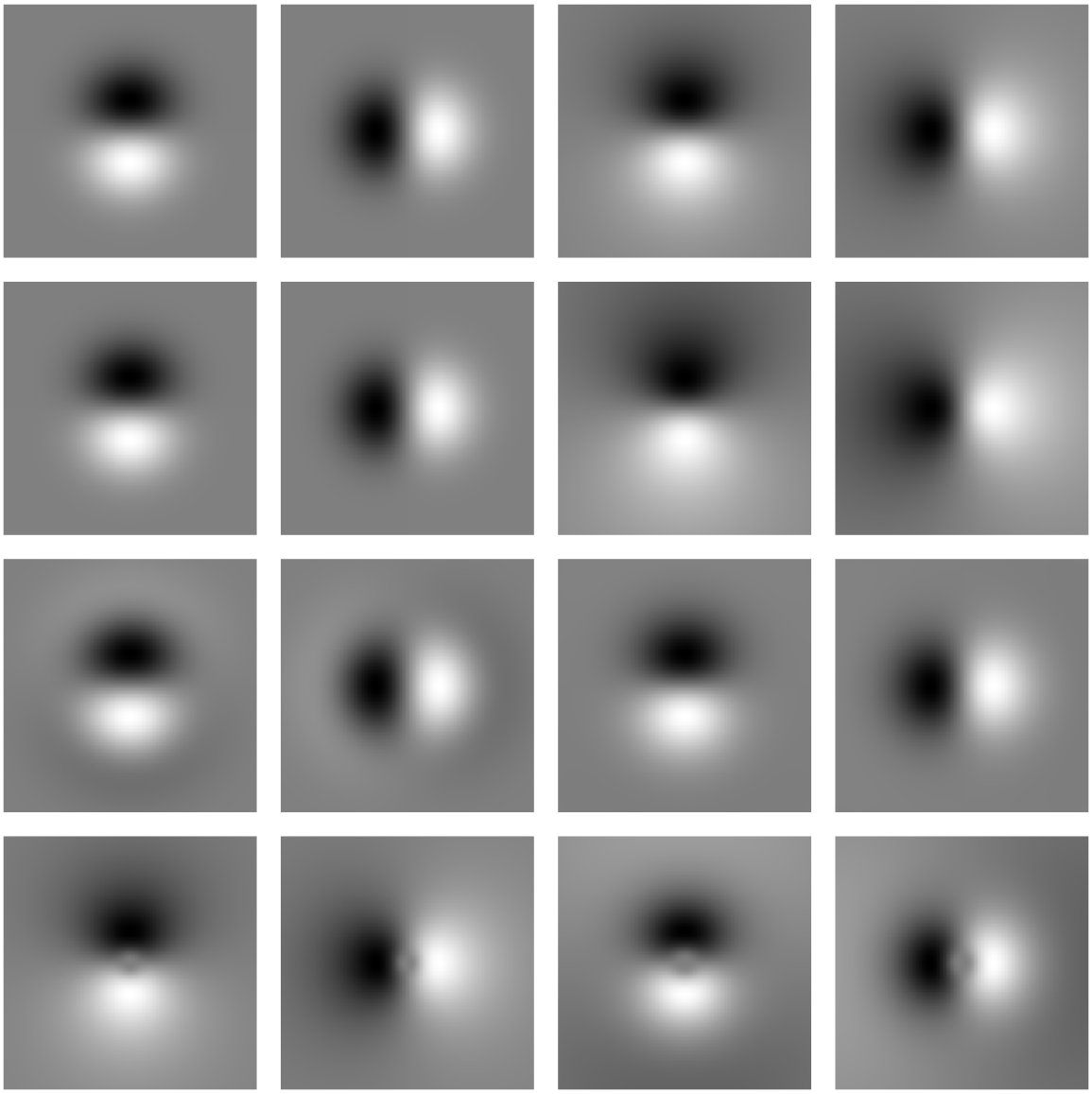
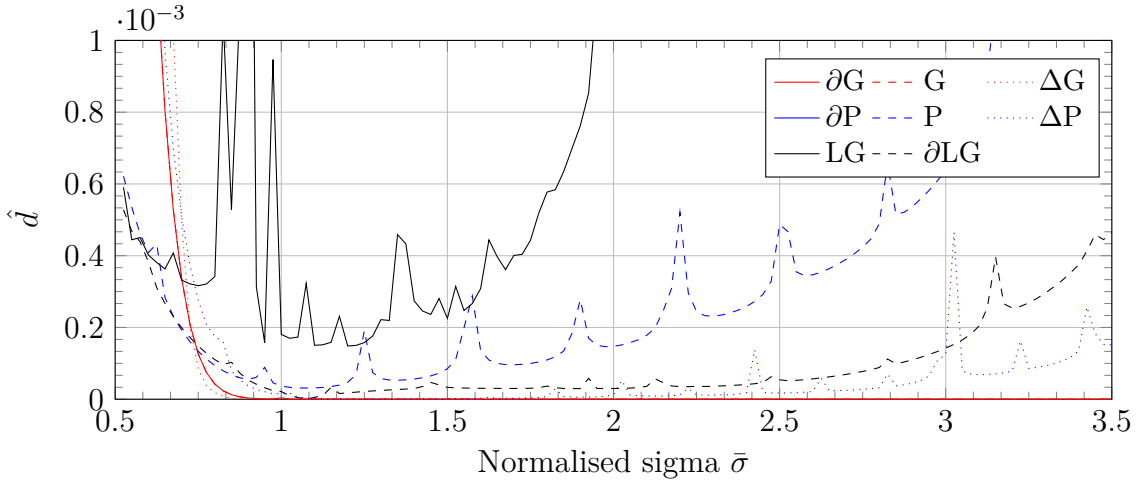


Figure 5.21.: Pairs of Spherical Quadrature Filters. Top row - $G(u, v, \sigma)$ and $P(u, v, \sigma)$. Second row - $\partial G(u, v, \sigma)$ and $\partial P(u, v, \sigma)$. Third row - $\Delta G(u, v, \sigma)$ and $\Delta P(u, v, \sigma)$. Forth row - LG and ∂LG.

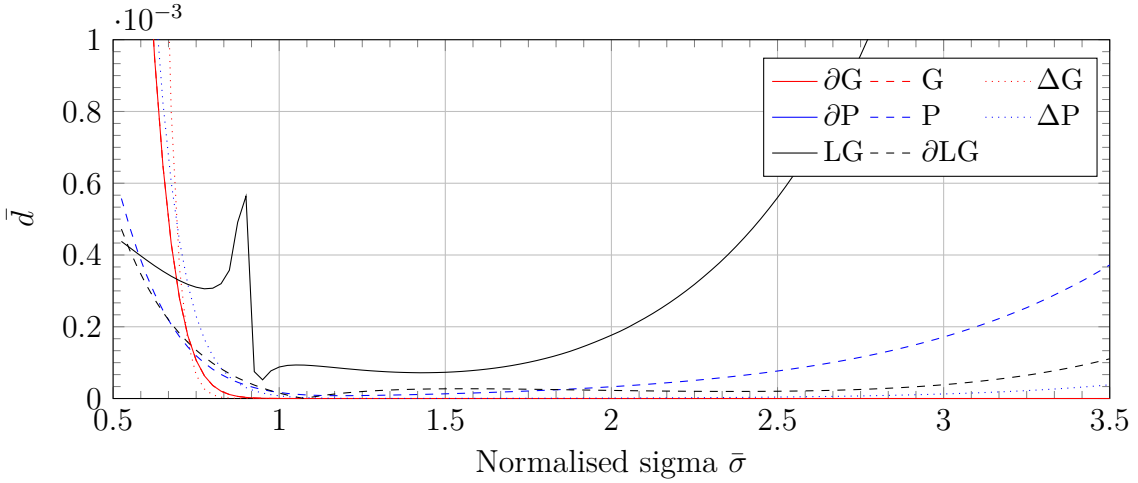
5.3.3. Accuracy of SQF's

The same methodology which was used to determine the accuracy of isomorphic kernels (section 5.2.3), is now performed for each of the SQFs. However, in this case the convolution between the filter and the image is performed using the *periodic FFT* [86] , rather than spatial convolution, in order to reduce the affects of ringing in the resultant image.

Figure 5.22 illustrates the average error of each SQF, with (a) and without (b) magnitude removal, when applied to the test image for the particular case of $\lambda = 10$ and over the range $\bar{\sigma} \in [0.5, 3.5]$. It is apparent that without low magnitude removal each of the filters (with the exception of the Log-Gabor) has a periodic increase in error. The period of the error cycle varies with ratio of the support of the kernel and the simulated ridge frequency.



(a) $\bar{d}$ error without magnitude removal



(b) $\bar{d}$ error with magnitude removal

Figure 5.22.: Average error $\bar{d}$ for SQF's on the test image for $\lambda = 10$. SQF's are:
∂G is the Derivative of Gaussian, G is the Gaussian,
ΔG is the Laplacian of Gaussian,
∂P is the Derivative of Poisson, P is the Poisson,
ΔP is the Laplacian of Poison and
LG is the Log-Gabor and ∂LG is the Derivative of Log-Gabor

Thus it is apparent firstly, the process of *low magnitude removal* has a dramatic effect upon the average error in all cases, secondly there are contiguous regions of low error (unlike the use of 3 by 3 which resulted in a single minima).

Graphically from figure 5.22 (b) and numerically from table 5.2, it can be seen that the family of Gaussian SQFs provide the lowest error. In addition, the region of lowest error also correspond to the range of friction ridge wavelengths. Figure 5.23 illustrates the average error of the Gaussian family of SQFs, where the smallest error is generated by Laplacian of Gaussian filter over the entire friction ridge range $\lambda = [6, 15]$.

The Laplacian of Gaussian filter (LoG) with $\bar{\sigma} = 2.6$ (i.e. the centre of the contiguous region) will be used as the SQF orientation operators in all further work.

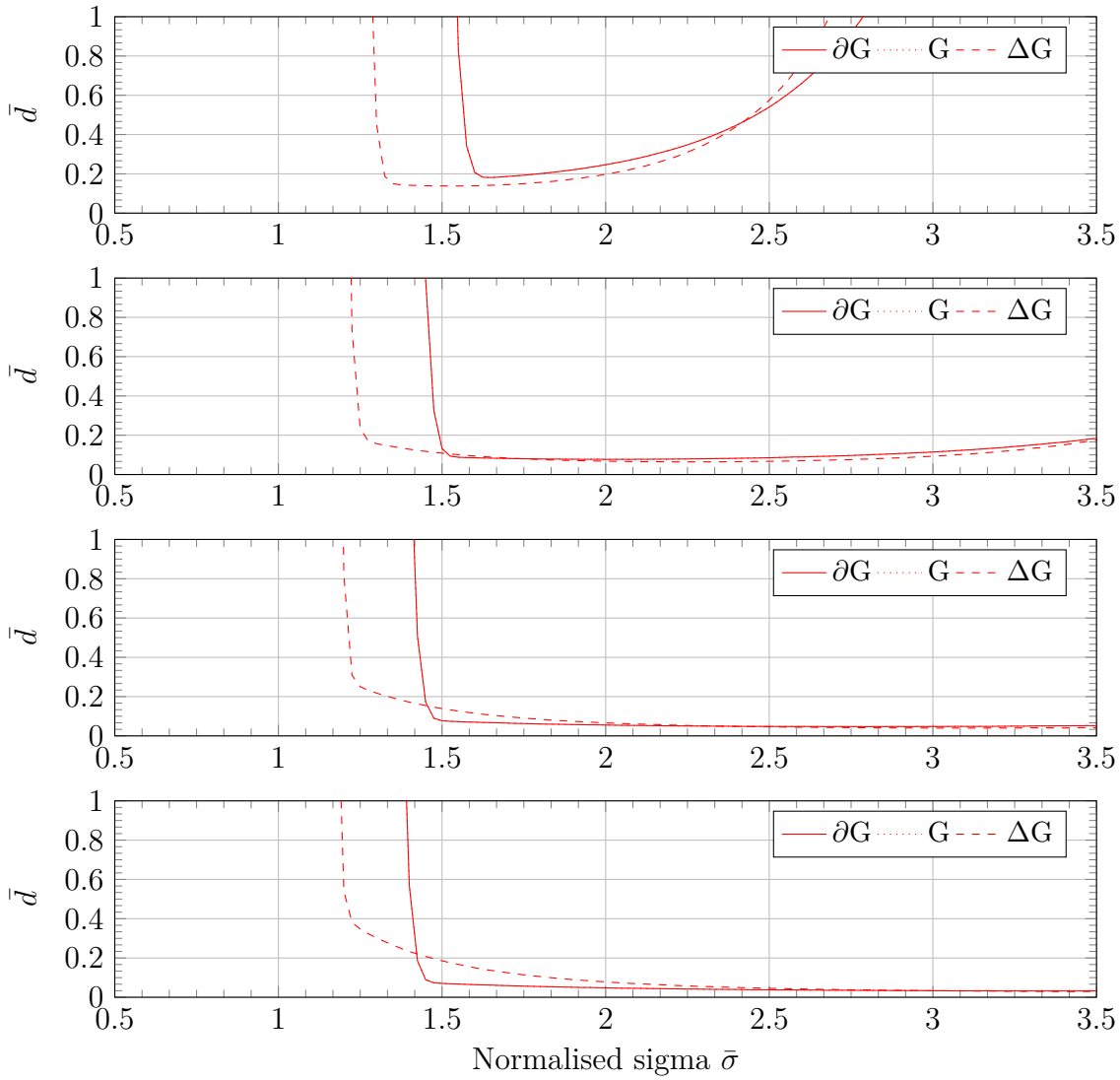


Figure 5.23.: Average error $\bar{d}$ (where $\bar{d} \in [0, 10^{-14}]$) for SQF's on the test image for $\lambda = \{6, 9, 12, 15\}$. SQF's are:
∂G is the Derivative of Gaussian, G is the Gaussian and
ΔG is the Laplacian of Gaussian

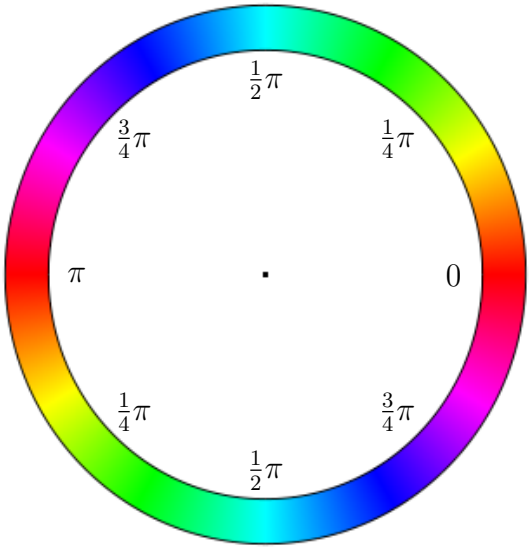
	λ			
SQF	6	9	12	15
Gaussian	1.8145×10^{-15}	7.7049×10^{-16}	4.6857×10^{-16}	3.1841×10^{-16}
Gaussian Derivative	1.8147×10^{-15}	7.1953×10^{-16}	4.7040×10^{-16}	3.1487×10^{-16}
Laplacian of Gaussian	$\mathbf{1.3809 \times 10^{-15}}$	$\mathbf{6.4420 \times 10^{-16}}$	$\mathbf{3.9538 \times 10^{-16}}$	$\mathbf{2.8952 \times 10^{-16}}$
Poisson	1.4582×10^{-3}	5.0780×10^{-4}	1.1712×10^{-3}	9.1794×10^{-4}
Poisson Derivative	2.6377×10^{-5}	1.9955×10^{-6}	5.3150×10^{-6}	3.6660×10^{-6}
Laplacian of Poisson	2.6377×10^{-7}	7.3663×10^{-8}	2.0444×10^{-7}	2.1343×10^{-7}
Log Gabor	1.2851×10^{-5}	1.0788×10^{-7}	2.8882×10^{-7}	2.1970×10^{-7}
Log Gabor Derivative	1.1210×10^{-5}	1.9927×10^{-5}	1.0278×10^{-5}	2.7532×10^{-6}

Table 5.2.: The smallest average error for the given SQF for $\lambda = \{6, 9, 12, 15\}$

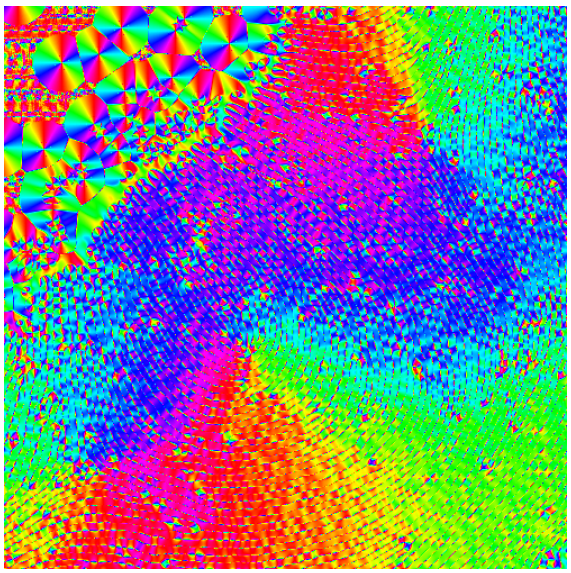
5.3.4. Low Magnitude Removal in Raw LoG Orientation Estimates of a palmprint

This section now describes and illustrates the affect of low magnitude removal upon the LoG orientation estimate of a palmprint, in which the unreliable estimates are replaced by NaN. Figure 5.24 illustrates the LoG orientation field of a tri-radiate region of a palmprint, in which low magnitude removal has been applied. The sub-figures in 5.24 illustrate: (a) the palmprint, (b) the orientation colour map, (c) the *raw* orientation field $\theta_{LoG}(x, y)$, (d) the LoG orientation field, $\bar{\theta}_{LoG}(x, y)$, with low magnitude removal.

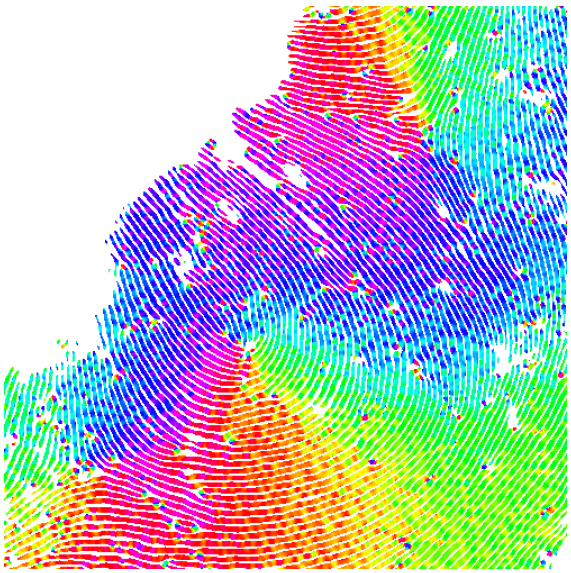
(a) Palmprint



(b) Axial orientation colour key



(c) Raw orientation field $\theta_{LoG}(x, y)$



(d) Orientation field with low magnitude removal $\bar{\theta}_{LoG}(x, y)$

Figure 5.24.: The affect of removing low magnitude orientation estimates on a Palmprint

By inspecting the palmprint in figure 5.24 (a) and $\bar{\theta}_{LoG}(x, y)$ in figure 5.24 (d) the following observations about the affect of low magnitude removal can be made:

- The large white area at the top left of $\bar{\theta}_{LoG}(x, y)$ corresponds to the empty space at the top left of the palmprint image. Therefore the valid palmprint region has been segmented from the background of the image. This affect can be exploited in later stages of APIS.
- Other smaller areas of NaN values do exist in $\bar{\theta}_{LoG}(x, y)$ and correspond to areas of the palmprint where the contrast is low.
- The minima and maxima of each friction ridge in $\bar{\theta}_{LoG}(x, y)$ can be identified as white lines (i.e NaN values), which are approximately parallel to one another (except around Level I structures). The coloured values represent the orientation of the edges of each friction ridge.
- As seen in section 2.3.5, minutiae points exist at the end points of each friction ridge. The minutiae are evident in $\bar{\theta}_{LoG}(x, y)$ as a change in colour; if one of the edges of a fiction ridge is followed, generally an edge will come to an *end* point, where the edge wraps around through π radians and joins up to the other side of the ridge. As this edge wraps around, the colour of of the edge varies through π in our colour map.

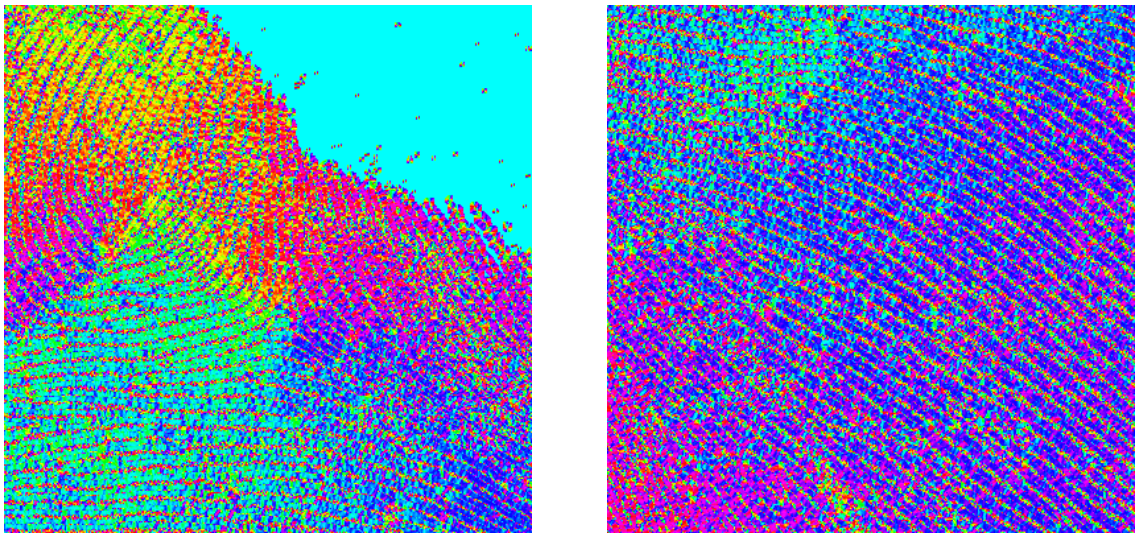
5.4. Summary

Figure 5.25 illustrates the orientation estimate for Sobel operator (as used in APIS [27] without low magnitude removal), the use of λ_{opt} and LoG, applied to two regions of the palm. The first region, contains two Level I structures (i.e. contains areas of high circular variance, in the left hand column). The second (in the right hand column), contains a region of the hypothenar which has low circular variance.

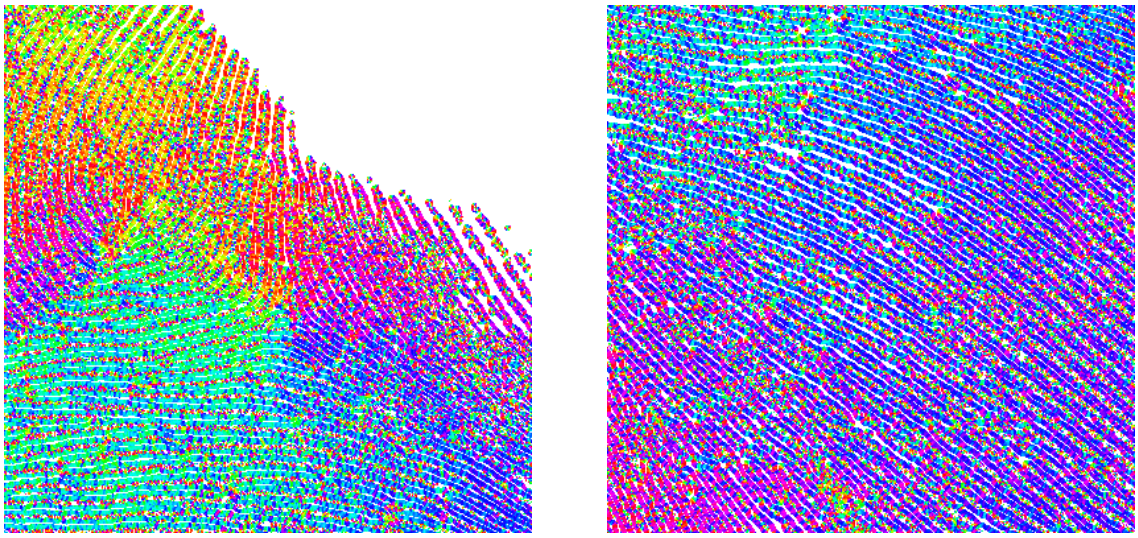
In figure 5.25, the rows illustrate the improvement of the orientation field described in this chapter. The top row illustrates a current technique [27] (FAA) in which the unreliable orientation estimates are left in the orientation field, this is seen visually as changes in the colour the minimas and maximas of the image and in the areas of low changes in intensity.

The middle row illustrates the orientation field calculated with λ_{opt} with low magnitude removal, which is an improvement upon current techniques. These improvements can be seen visually by the removal of the unreliable orientation estimates (the white areas).

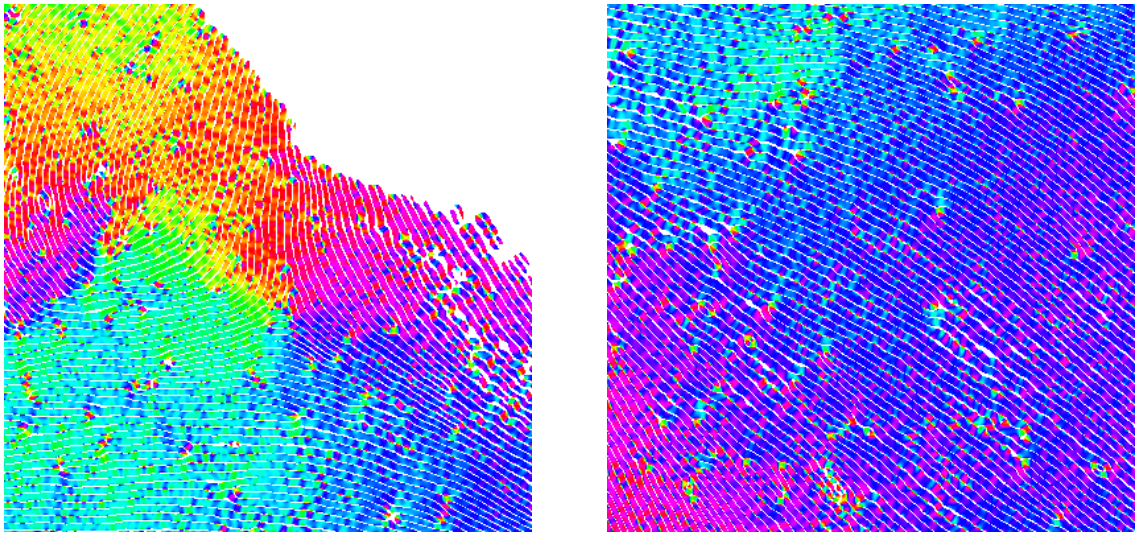
The bottom row illustrates the orientation field calculated with LoG with low magnitude removal, which shows further improvements over the current technique and the λ_{opt} technique. These improvements can be seen visually by the consistency of the orientation estimates in a local area.



(a) FAA



(b) $\lambda_{opt}$with low magnitude removal



(c) LoG with low magnitude removal

Figure 5.25.: The left column illustrates the orientation operators on a region of the palm containing 2 Level I structures, the column on the right illustrates the orientation operators on a relatively low variance

5.4.1. Extraction of the Raw Axial LoG Orientation Estimate of a Palmprint

This section provides a summary of the steps used in this thesis, to extract the orientation estimates from a palmprint. The result of each step are shown visually in figure 5.26.

1) An estimation of the gradients are calculated using the normalised LoG filters with $\bar{\sigma}_{fr} = 2.6$ as below, producing the *gradients*, $F_x(x, y)$ and $F_y(x, y)$

$$\begin{aligned} F_x(x, y) &= I(x, y) \oplus LoG_u \\ F_y(x, y) &= I(x, y) \oplus LoG_v \end{aligned} \tag{5.32}$$

2) The gradients are then used to calculate the *raw orientation field* $\theta_{LoG}(x, y)$ and *magnitude* $\alpha(x, y)$:

$$\theta_{LoG}(x, y) = \left(\frac{\pi}{2} + \frac{1}{2} \text{atan2}\left(F_y(x, y), F_x(x, y)\right) \right) \text{mod} \pi \in [0, \pi) \tag{5.33}$$

$$\alpha(x, y) = \sqrt{F_u(x, y)^2 + F_v(x, y)^2} \tag{5.34}$$

3) The magnitude $\alpha(x, y)$ is normalised between $[-1, 1]$to produce $\overline{\alpha}(x, y)$:

$$\overline{\alpha}(x, y) = \frac{\alpha(x, y) - \min(\alpha(x, y))}{\max(\alpha(x, y)) - \min(\alpha(x, y))} \in [0, 1] \tag{5.35}$$

4) Low magnitude components ,$\overline{\alpha}(x, y) < 0.5$, are removed from the orientation field $\theta_{LoG}(x, y)$, resulting in $\bar{\theta}_{LoG}(x, y)$.

$$\bar{\theta}_{LoG}(x, y) = \begin{cases} \theta_{LoG}(x, y) & \text{if } \overline{\alpha}(x, y) > \frac{1}{2} \\ NaN & \text{if } \overline{\alpha}(x, y) \leq \frac{1}{2} \end{cases} \tag{5.36}$$

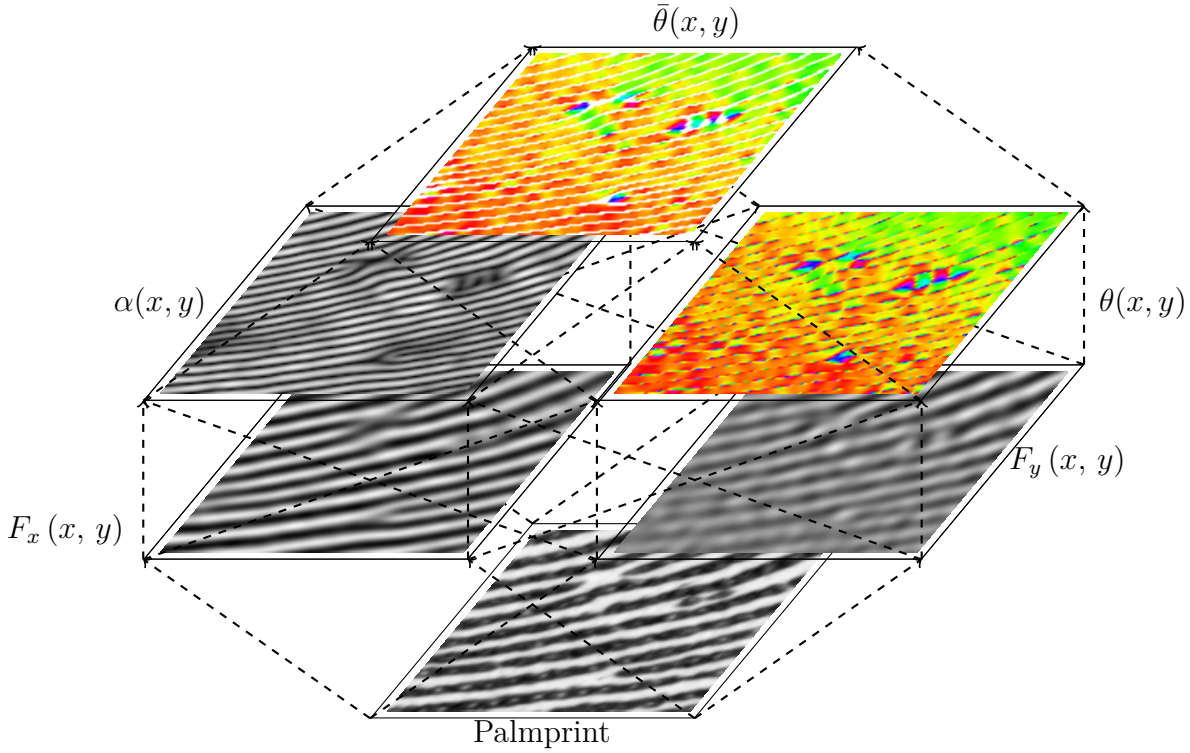


Figure 5.26.: The process of extracting the orientation from a Palmprint

5.5. Conclusions

It has been shown that there is a strong relationship between the scale of the convolution filter (the value of b for 3 by 3 isometric kernels and the value of $\bar{\sigma}$ for SQF's) and the wavelength of the objects in an image.It has been further shown that the scale can be chosen to optimise the accuracy of the resulting orientation estimates.

As a result, a new 3 by 3 kernel, λ_{opt}, has be developed , which has been shown to provide lower average error than a selection of 3 by 3 isometric kernels, currently used in APIS/AFIS.

As result of the development of the optimised SQF, it has been shown that the LoG filter has the lowest average error and that, unlike the isometric kernels, this occurs over a range of frequencies (rather than occurring at a single minima) and that this range corresponds to the range of friction ridge wavelengths.

All estimates contain unreliable values. Current APIS and AFIS do not remove unreliable orientation estimates, rather they use some form of averaging technique which smooths the unreliable orientation estimates and so distorts the neighbouring orientation field. The resulting distortions limits the use of higher statistics on the resulting image.

This chapter has developed a novel method of identification and removal of these values. The removal of the unreliable values has been shown to also segment the palm print from its background and will be shown to admit the calculation of reliable higher order circular statistics.

However, the removal of the unreliable estimates leaves gaps within the orientation field. The following chapter describes a region growing method which is used to interpolate

within the gaps.

Chapter 6

Orientation Estimate Filtering

6.1. Introduction

As discussed in chapter 3, the purpose of the orientation field in current APIS and AFIS is to provide an estimate of the *orientation of the friction ridges* in a *local* area (for use in contextual filtering) whilst suppressing Level II and III detail. The estimate is regarded as the orientation of Level I detail. The orientation values, created by current APIS, do not represent the orientation of a particular pixel, but of the *average orientation* of the pixels contained within a square region. This averaging process reduces the affect of the unreliable estimates (i.e. Level II and Level III features), however it also distorts the orientation field, particularly around areas of high curvature.

The process of enhancement described in the previous chapter, takes an alternative approach, by removing unreliable orientation estimates, rather than retaining them. The removal of unreliable estimates introduces gaps in to the orientation field, which must be replaced.

This chapter describes a iterative region based algorithm, which replaces the gaps within the orientation estimates using a weighted neighbourhood estimate. The weights are derived from both the magnitude and circular mean of the neighbourhood. The algorithm has the additional advantage of suppressing Level II and III detail.

The final orientation estimate is compared with a current APIS technique, where it is shown that less of the level II and III detail remains. The effect of the improvement in the orientation estimate is demonstrated, by replacing the orientation estimation method in an APIS technique and showing an improvement in minutiae identification (section 6.4).

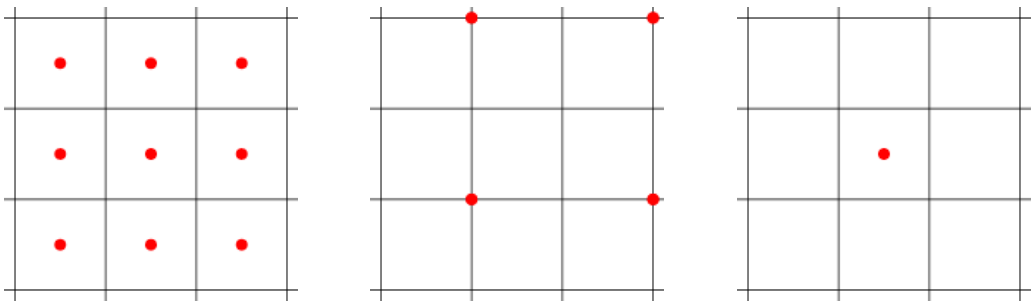
The structure of the chapter is as follows: Section 6.2 introduces methods for the calculation of circular mean orientation, variance, skew and kurtosis within local circular regions of the image. The section ends with examples of the statistics applied to the orientation field and indicates how both regions of Level I structures and detail will be isolated in Chapter 6.3. Section 6.2 is a necessary precursor prior to the description of the adaptive filtering algorithm in section 6.3. Section 6.4 contains a comparison of the orientation estimate with other APIS techniques. The chapter concludes with a demonstration of how the filtered orientation field can be used to improve current APIS techniques.

6.2. Local Image Analysis using Circular Statistics

6.2.1. Introduction

This section illustrates, how *local* circular statistics can be calculated over relatively small circular regions, in order to analyse the local circular properties of an image. The circular regions are centred at points specified by their separation, n, and a given radius, r. If n is odd, the centre of the local region lies at the *centre of a pixel* and if n is even, the centre of the local region lies at the *centre of four 4-connected* pixels. The centre of local regions are illustrated by the red dots in figure 6.1, for different values of n.

The algorithm 6.1 (LSC) can be used to calculate circular statistics for each pixel. However, if it is required that the resultant image be of a lower resolution than the



(a) $n = 1$ (b) $n = 2$ (c) $n = 3$

Figure 6.1.: Boxes represent the pixel boundaries, the red dot represents the central point of each local region

original (i.e. de-scaled), this can be achieved by choosing $n > 1$. Hence, n controls the scaling factor of the final *image*.

6.2.2. Method

Algorithm 6.1 describes how, a *local* weighted circular mean $\bar{\theta}_{n,r}$, the circular variance $\bar{R}_{n,r}$, skew $\bar{s}_{n,r}$ and kurtosis $\bar{k}_{n,r}$, of an image $I(x, y)$, can be calculated, given the *axial orientation* $\theta(x, y)$ and the associated magnitude $\alpha(x, y)$. The circular regions are created by multiplying a square neighbourhood (lines 1-2) by a mask, which excludes any pixel which is further than r from the centre neighbourhood. Two distinct masks are required for the cases, when n is odd and when n is even. The masks are illustrated in figure **??** and calculated in lines 4-5. The pixels within a local region may also be weighted using a mask (line 9).

The calculation of the circular statistics takes place in lines 11-18, where it should be noted that each pixel is *implicitly weighted* according to the value of its magnitude (i.e its reliability) in all calculations.

Algorithm 6.1 LCS - Local circular statistics of an image

Require: $\theta(x,y)$, $\alpha(x,y)$, n and r
Ensure: $\bar{\theta}_{n,r}(x_d,y_d)$, $\bar{R}_{n,r}(x_d,y_d)$, $\bar{s}_{n,r}(x_d,y_d)$ and $\bar{k}_{n,r}(x_d,y_d)$,
$x_d = 1 : (x - x\text{mod}n)/n$, $y_d = 1 : (y - y\text{mod}n)/n$
Create empty arrays: $\bar{\theta}_{n,r}(x_d,y_d)$, $\bar{\alpha}_{n,r}(x_d,y_d)$, $\bar{s}_{n,r}(x_d,y_d)$ and $\bar{k}_{n,r}(x_d,y_d)$
if n is even **then**
    Create arrays such $[u,v] \in R^2$ of dimension (2r,2r)
    else Create arrays such $[u,v] \in R^2$ of dimension (2r+1,2r+1)
end if
for i in $1 : x_d$ **do**
    for j in $1 : y_d$ **do**
        Create arrays $\theta_{tmp} = \theta(u+in, v+jn)$, $\alpha_{tmp} = \alpha(u+in, v+jn) \times \text{mask}$
        if length $\theta_n > lim$ **then**
            $\bar{\alpha} = \sum \alpha_p$
            $\hat{a} = \frac{1}{\bar{\alpha}} \sum \alpha_p \cos 2\theta_p$
            $\hat{b} = \frac{1}{\bar{\alpha}} \sum \alpha_p \sin 2\theta_p$
            $\hat{\theta} = \text{atan2}(\hat{b}, \hat{a})$
            $\bar{\theta}_{n,r}(i,j) = (\frac{\pi}{2} + \hat{\theta}) \text{mod} \pi$
            $\bar{R}_{n,r}(i,j) = \frac{1}{\bar{\alpha}} \sqrt{\hat{a}^2 + \hat{b}^2}$
            $\bar{s}_{n,r}(i,j) = \frac{1}{\bar{\alpha}} \sum 2\alpha_{tmp} \cos(\theta_{tmp} - \hat{\theta}) \sin(\theta_{tmp} - \hat{\theta})$
            $\bar{k}_{n,r}(i,j) = \frac{1}{\bar{\alpha}} \sum \alpha_{tmp} (\cos^2(\theta_{tmp} - \hat{\theta}) - \sin^2(\theta_{tmp} - \hat{\theta}))$
        end if
    end for
end for

6.2.3. Examples of Local Image Analysis using Circular Statistics

Figure 6.2 illustrates the the application of the algorithm to part of a palmprint. Figure 6.2 (a) illustrates the raw orientation field $\theta(x, y)$ overlaid upon a palmprint, where the magnitude $\alpha(x, y)$ is represented by the length of the segments and the orientation by the colour and direction of the segment. Figure (c) and (d) illustrates the orientation data as circular histograms for each of the regions in figure (a). Figure (c) illustrates the raw orientation field on a ridge and figure (d) illustrates the orientation field about a minutiae.

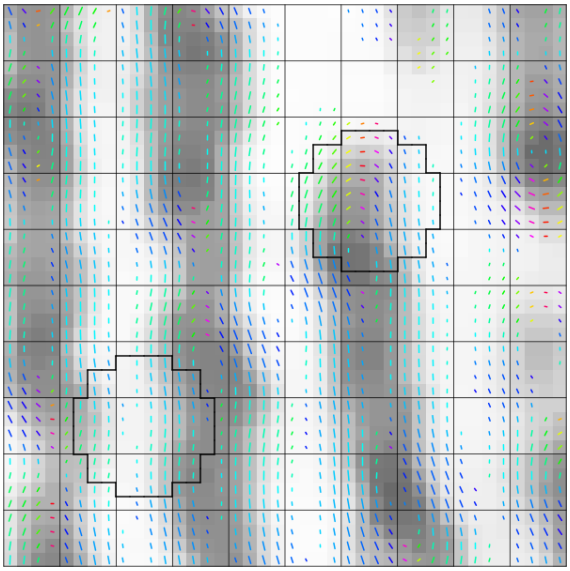
Figure 6.2 (b) illustrates results of the algorithm where the *local* circular mean $\bar{\theta}_{n,r}$ is overlaid upon the original palmprint and where the length of the segment represents the circular variance $\bar{R}_{n,r}$.

The plots in 6.2 (c) and (d) are illustrated because they show two different situations.

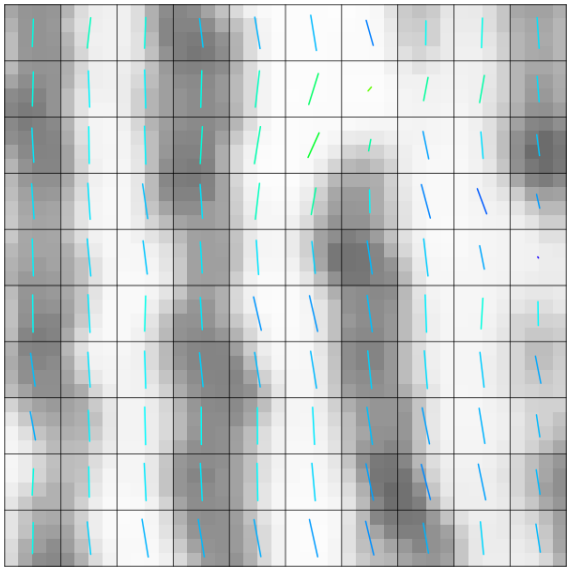
Figure 6.2 (c) illustrates an archetype friction ridge region that has relatively minor deviations in orientation. This is illustrated by the regions orientations being grouped together in the pale blue region, around $\frac{\pi}{2}$. Hence the *mean resultant length* of the region is relatively high, the value is approximately 0.95 and is represented by the length of the corresponding resulting segment in figure 6.2 (b).

Figure 6.2 (d) illustrates an archetype friction ridge region near a minutiae point, hence the local region has relatively large deviations in orientation. This is illustrated by the regions orientations being spread throughout the circular domain. Hence the *mean*

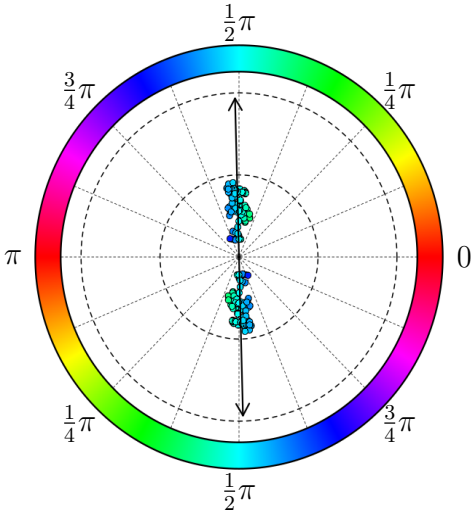
resultant length of the region is relatively low, the value is approximately 0.6, which is represented by the length of the corresponding resulting segment in figure 6.2 (b).



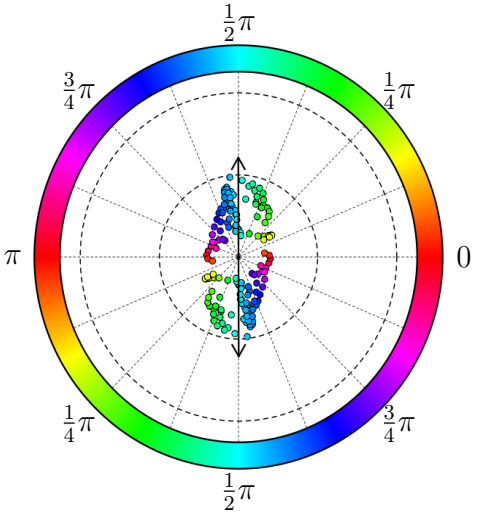
(a) Palmprint where the orientation is overlaid as segments (the length of a segment indicates its magnitude) and two local regions.



(b) Palmprint where the local circular ($n = 4$, $r = 5$) orientation is overlaid as segments (the length indicates its magnitude)



(c) Circular statistics of the orientation data indicated by the bottom left region in (a)



(d) Circular statistics of the orientation data indicated by the top right region in (a)

Figure 6.2.: Local circular statistics

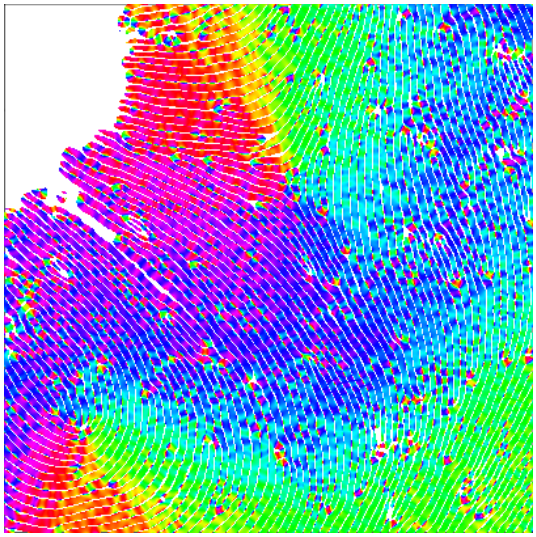
Figure 6.3 illustrates the properties of the variance, $\bar{R}_{n,r}$, skew, $\bar{s}_{n,r}$, and kurtosis, $\bar{k}_{n,r}$, of the local circular statistics on a palmprint.

Figure 6.3 (c) to (f) are the results of local circular statistics, performed on the palmprint in (a). Figure 6.3 (b) is the orientation field from (a) using the low magnitude removal method $\overline{\theta}(x, y)$ from section 5.3, then the local circular statistics are calculated using $\overline{\theta}(x, y)$ and $\overline{\alpha}(x, y)$, where $n = 4$ and $r = 5$.

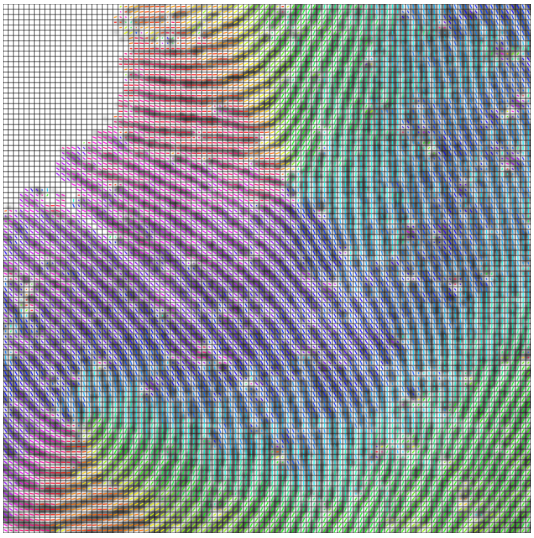
The local orientations $\bar{\theta}(x_d, y_d)$ in 6.3 (c) is a representation of the local mean circular direction of the features within an image, where each segment's length represents the

local variance $1 - \bar{R}\,(x_d, y_d)$.

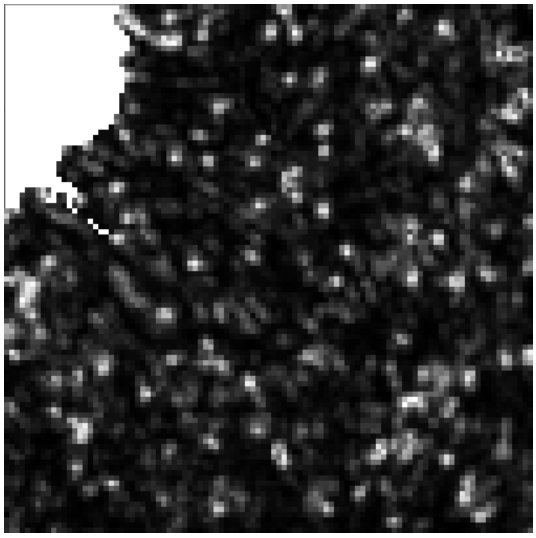
(a) Palmprint $I(x, y)$



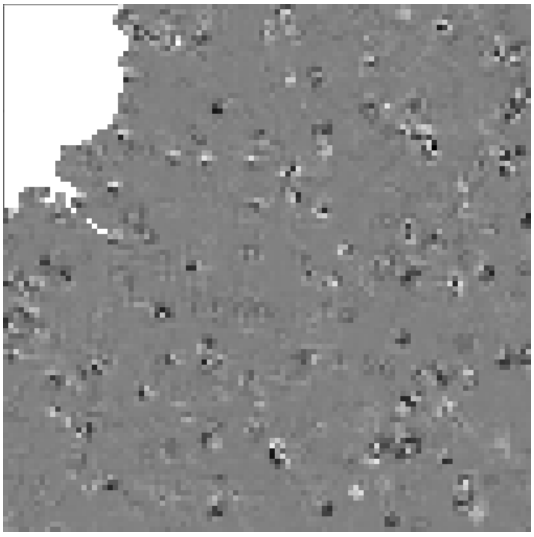
(b) Orientation of (a) $\theta(x, y) \in [0, \pi)$



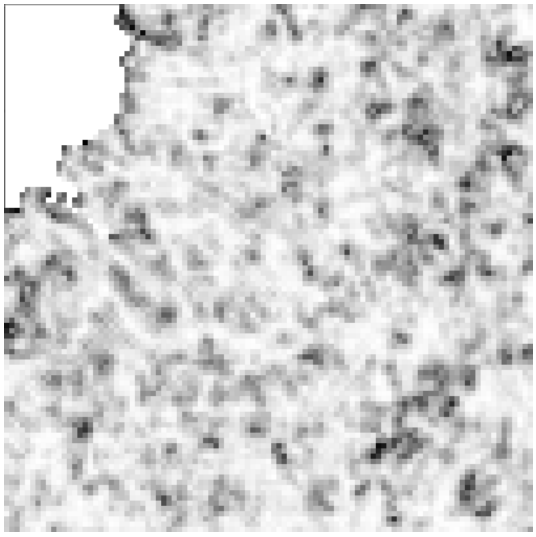
(c) Local Orientation $\theta(x_d, y_d) \in [0, \pi)$



(d) local Variance $\bar{R}(x_d, y_d) \in [0, 1]$



(e) local Skew $\bar{s}(x_d, y_d) \in [-1, 1]$



(f) Local Kurtosis $\bar{k}(x_d, y_d) \in (-\infty, 1]$

Figure 6.3.: Local circular statistics, where $n = 4$ and $r = 5$

6.3 (d) illustrtes the local variance $1 - \bar{R}(x_d, y_d)$ where the darker areas indicate the friction ridge flow is of a similar orientation, whereas the lighter areas indicate where the friction ridge flow is dissimilar . The lighter areas include: Level I structures and minutiae.

The local skew $\bar{s}(x_d, y_d)$ shown in 6.3 (e) has areas of middle grey indicating where the local friction ridge flow from (a) is approximately symmetrical about the circular mean. The lighter areas indicate where the friction ridge flow is skewed clockwise about the circular domain. The darker areas indicate where the friction ridge flow is skewed counterclockwise about the circular domain.

The local kurtosis $\bar{k}(x_d, y_d)$ in 6.3 (f) can take on very large negative values when the circular data is spread around the circular domain. The lighter areas indicate where the local friction ridge flow from (a) is relatively tightly distributed about the circular domain. The darker areas indicate where the friction ridge flow is scattered about the circular domain.

6.3. Adaptive Local Neighbourhood Filtering

The following section describes how the observations can be used for removing any estimates which still contain any Level II and II detail, then filling the empty values to improve the final orientation estimate.

As stated in the introduction, the function of orientation field filtering is two fold. Firstly to suppress unwanted detail, in this case Level II and III. Secondly to calculate new orientation estimates for those unreliable and unwanted estimates, which have been removed. This is achieved using the ALNF (*Adaptive Local Neighbourhood Filtering*) algorithm, shown in algorithm 6.2.

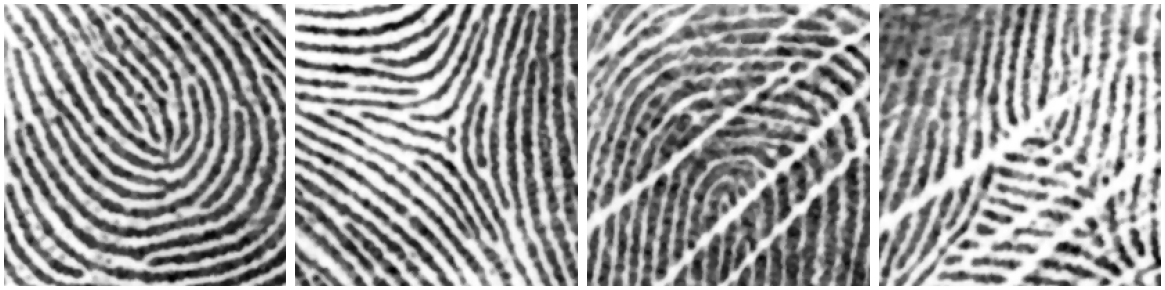
6.3.1. Overview

Steps 2-4 of algorithm 6.2 suppress the effect of unwanted detail in the orientation image using properties of image's circular variance and skew. These properties are used to form masks which when applied to the orientation only select the required pixels. At this stage all unreliable and unwanted estimates have been removed. This reduces the affect of distortion around level I structures, and reduces or eliminates the undesirable orientation estimates produced by level II and Level III features.

Steps 5-6 generate replacement estimates by growing regions pixel by pixel within a neighbourhood, whose membership is controlled by the circular variance. It will be shown that the strict membership criterion used during these stages limits growth in areas of similar orientation.

Steps 7-9 The circular mean, weighted by its variance, is applied recursively to the orientation estimates, so that growth is no longer limited to regions of similar orientation.

To illustrate the adaptive local neighbourhood filtering method, the palmprint images in figure 6.4 will be used to show affect of each stage. The palmprints have been chosen so that the figure 6.4 (a) contain a single delta, figure 6.4 (b) a single loop, whilst figure 6.4 (c) and (d) are used illustrate the effect of orientation estimates when creases occur upon a loop and a vestige. The latter cases have been explicitly included since it is known that current APIS fail in such cases [39].



(a) Loop (b) Delta (c) Loop with creases (d) Vestige with creases

Figure 6.4.: Parts of palmprint used for illustrating adaptive local neighbourhood filtering. (a) and (b) are typical low varinace regions, (c) and (d) illustrate ares where current APIS are inaccurate

Algorithm 6.2 ALNF - Adaptive local neighbourhood filtering

Require: $\theta(x,y)$, $\alpha(x,y)$
Ensure: $\hat{\theta}$
1: LCS= $LCS(\theta(x,y),\, \alpha(x,y), n=4, r=3)$
2: $mask_{\bar{R}}$ =LCS$_{\bar{R}} > 0.75$
3: $mask_{\bar{s}}$ =abs(LSC$_{\bar{s}}$) < 0.05
4: $\hat{\theta} = (\hat{\theta} = (mask_{\bar{s}} \cap mask_{\bar{R}}) \cdot \bar{\theta}_{LCS}$ ▷ Removes high $1-\hat{R}$ and $\bar{s}$ values
5: $\hat{\theta}$ =G4C($\hat{\theta}$)
6: $\hat{\theta}$ =GLCS($\hat{\theta}$)
7: **for** $i = 1:5$ **do**
8: $\hat{\theta} = LCS(\hat{\theta},\, 1-\hat{R}, n=1, r=3)$ ▷ Weighted where $i \geq 2$
9: **end for**

6.3.2. Detailed Overview and Example on Level I Structures

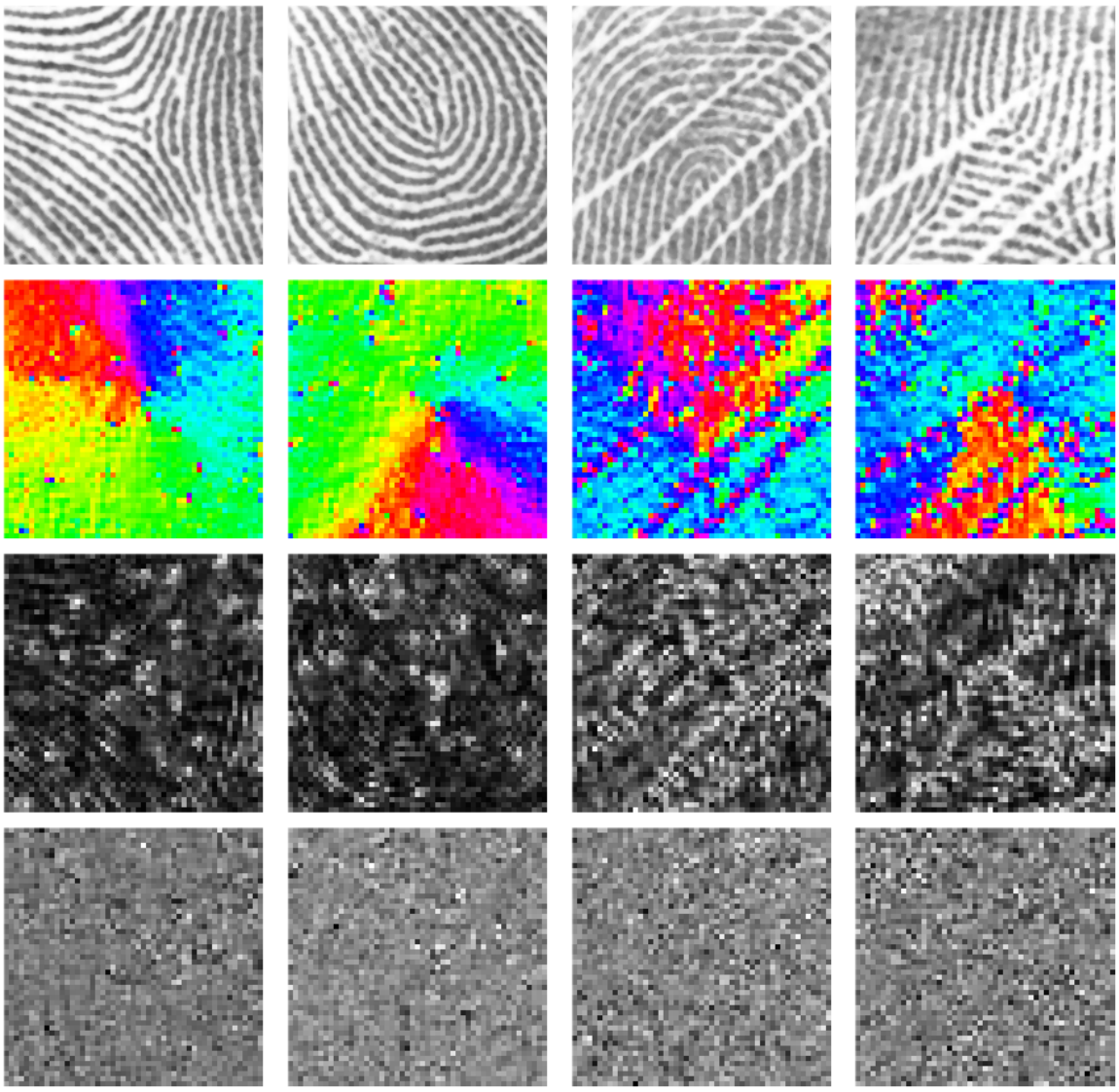
Each step of algorithm 6.2 will now be described and illustrated.

Step 1

The purpose of step 1 is to analyse the orientation field with the aim of removing any orientation estimates that occur due to Level II and Level III features.

The local circular statistics are generated using the LCS algorithm for $n = 4$ and $r = 3$. Therefore, the image used in further steps has been de-scaled by a factor of 4. Figure 6.5 illustrates the statistics for palmprint.

It is apparent that the effects of level II detail (i.e. minutiae) and Level III (i.e. creases) appear in the corresponding statistics. In the variance field they appear as the higher values (i.e lighter), which are relatively noisy about the creases. The direction of the creases in both orientation and skew fields at is approximately $\frac{\pi}{4}$, indicted by dark blue/purple colour line in the orientation field. The skew in the fourth row is difficult to interpret at this scale, but step 3 will show that creases and minutiae result in higher values of skew.



(a) Loop (b) Delta (c) Loop with creases (d) Vestige with creases

Figure 6.5.: LCS for parts of palmprints. Top row: Palmprint, Second row: Orientation field, Third row: Variance (1-$\bar{R}$) and Fourth row: Skew.

Step 2, 3 and 4

These steps, now remove high variance and high skew orientation estimates, which are indicative of Level II and Level III detail, using a simple threshold value.

Using a threshold on both the variance $\bar{R}_{LCS}$ and skew $\bar{S}_{LCS}$ separately, unwanted orientation estimates from $\bar{\theta}_{LCS}$ can be eliminated.

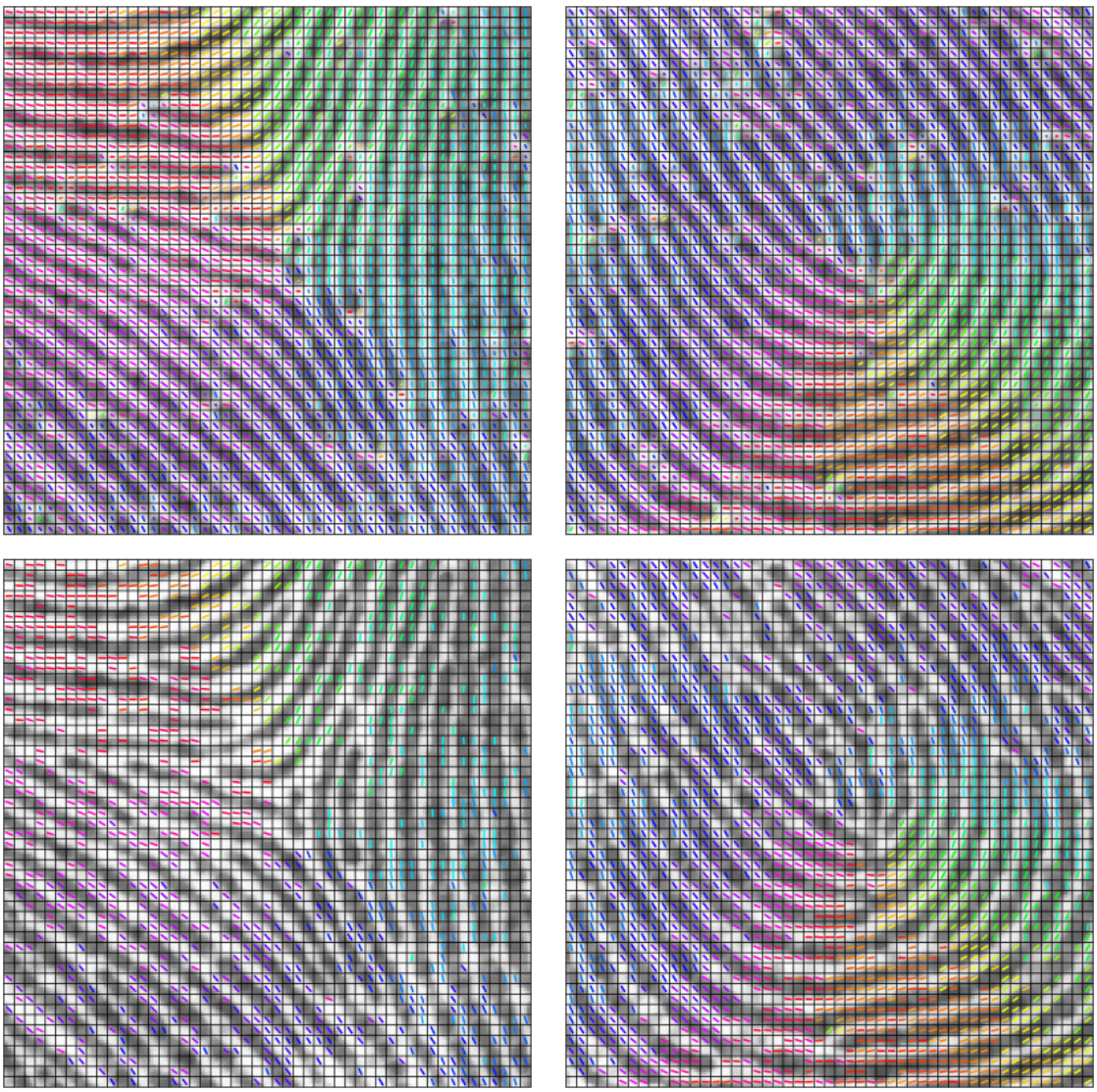
Tests have shown that a variance threshold of $\bar{R}_{LCS} > 0.75$ and skew threshold of $\bar{s}_{LCS} < 0.05$ in equation 6.1 can be used to produce $\hat{\theta}$, which removes a large proportion of the orientation estimates that arise as a result of Level II and Level III features in the orientation field. Figures 6.6 and 6.7 illustrate the effects of this process.

$$mask_{\bar{R}}(x, y) = \begin{cases} NaN & \text{if} \quad \bar{R}_{LCS}(x, y) > 0.75 \\ 1 & \text{if} \quad \bar{R}_{LCS}(x, y) \leq 0.75 \end{cases}$$

$$mask_{\bar{s}}(x,\, y) = \begin{cases} NaN & \text{if} \quad |\bar{s}_{LCS}| < 0.05 \\ 1 & \text{if} \quad |\bar{s}_{LCS}| \geq 0.05 \end{cases}$$

$$\hat{\theta} = (mask_{\bar{s}} \cap mask_{\bar{R}}) \cdot \bar{\theta}_{LCS} \tag{6.1}$$

In figure 6.6 approximately 20% of the orientation estimates are removed and correspond to regions where minutiae exist and where the orientation of the friction ridges is inconsistent.

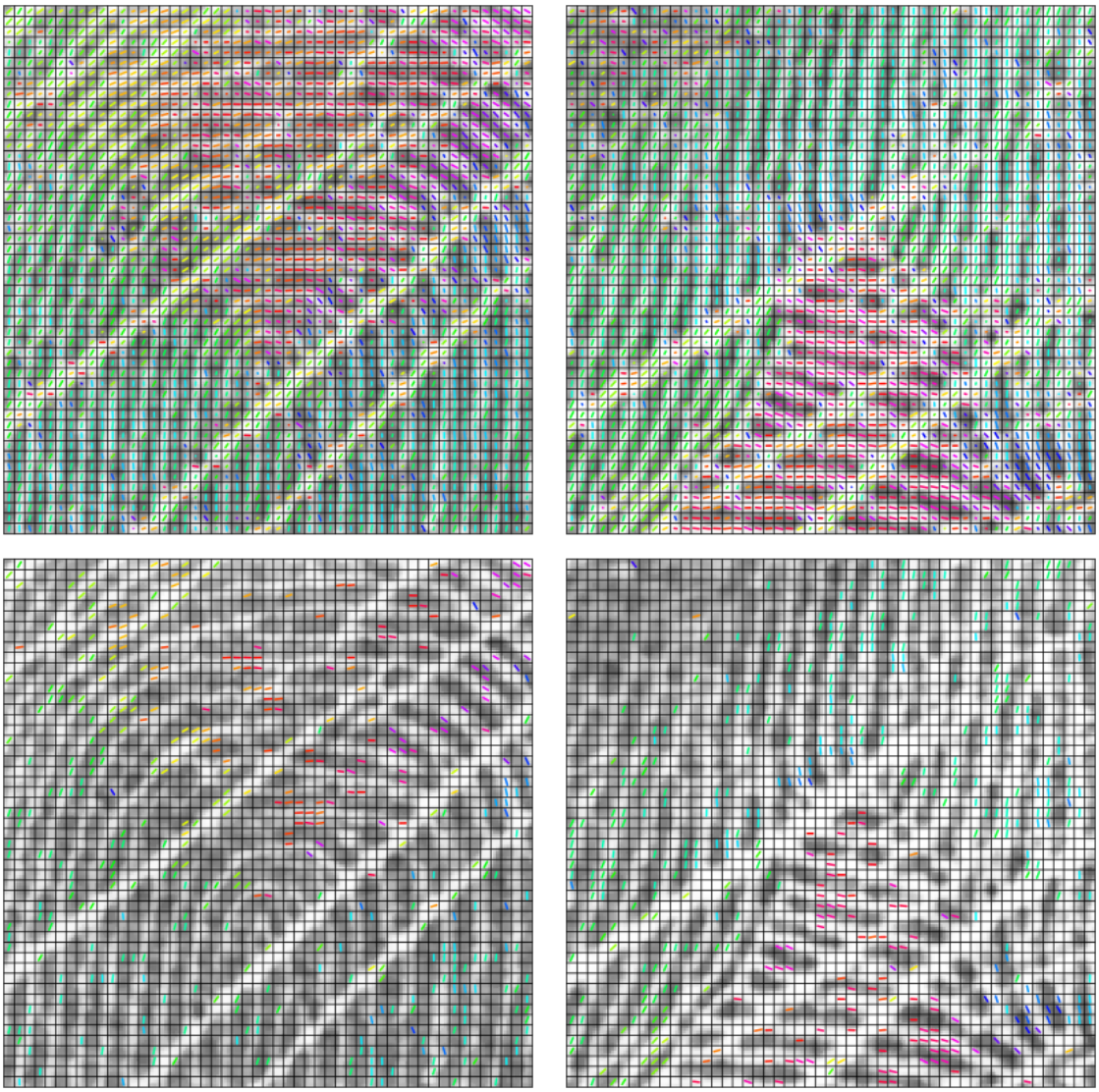


(a) Loop (b) Delta

Figure 6.6.: Removing orientation estimates. The orientation field from step 1 is illustrated on the top row and the orientation field with the removed values is illustrated on the bottom row.

In figure 6.7 approximately 80% of the orientation estimates are removed. Few estimates remain in the orientation field in figure 6.7 that are as a result of Level II and Level III features. Intuitively, it may appear too many orientation estimates have

been removed from these regions. However, the empty regions of the orientation field are now grown (in steps 5 and 6), to produce an orientation field that represents the orientation of the friction ridges.



(a) Loop with creases (b) Vestige with creases

Figure 6.7.: Removing orientation estimates

Step 5 and 6

The orientation field, $\hat{\theta}$ is now grown, using two region growing algorithms, *Growing 4 connected* (G4C) and *Growing from local circular statistics* (GLCS). GLCS operates of the results of G4C.

G4C (see algorithm 6.3) is an iterative process operating upon $\hat{\theta}$. During each iteration, a NaN is replaced by the circular mean of its 4-connected neighbours ,if

1. the NaN is connected to at least 3 of the 4-connected neighbouring pixels *and*
2. the circular variance of the NaN's 4-connected pixels is below a specified threshold.

The stopping criteria is that a complete iteration must have occurred without NaN replacement.

GLCS (see algorithm 6.4) applies iterations of LCS to the result of G4C (i.e. $\hat{\theta}$), for a small region of $r = 1$ and for $n = 1$. If central pixel of the region is a NaN and the region has a variance below a specified threshold, the central pixel is replaced by the circular mean of the region. The stopping criteria is that a complete iteration must have occurred without NaN replacement.

Algorithm 6.3 G4C - Growing 4 connected

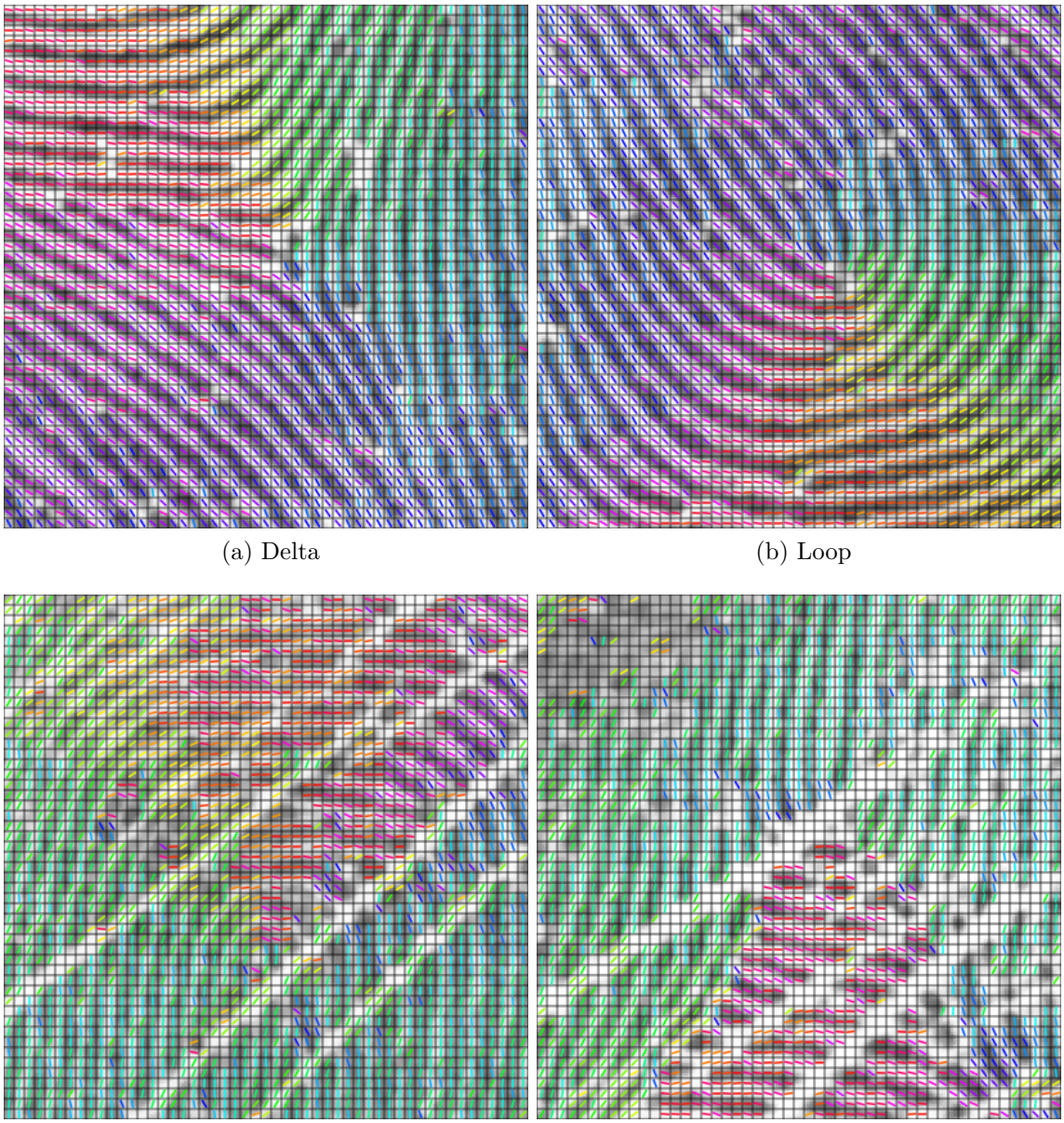
Require: $\theta(x, y)$
Ensure: $\theta(x, y)$
$p = 0$
while sum(valid($\theta(x, y)$))≠p **do**
    mask(x, y)=4C($\theta(x, y)$) ▷ number of valid 4 connected components
    for i in 1:x **do**
        for j in 1:y **do**
            if mask(i, j)> 2 **then**
                $\theta_n = \theta(i + 1, j + 1), \theta(i + 1, j - 1), \theta(i - 1, j + 1), \theta(i - 1, j - 1)$
                $a = \cos(\theta_n)$
                $b = \sin(\theta_n)$
                if $\frac{1}{|\theta_n|}\sqrt{a^2 + b^2} > 0.999$ **then**
                $\theta(i, j)$ =atan2(b, a)modπ
                end if
            end if
        end for
    end for
    p=sum(valid($\theta(x, y)$))
end while

Algorithm 6.4 GLCS - Growing from local circular statistics

Require: $\theta(x, y)$
Ensure: $\theta(x, y)$
LSC= $LSC(\theta(x, y), n = 1, r = 1)$ ▷ Unweighted
mask(x, y)=empty($\theta(x, y)$) ▷ Empty pixels in $\theta(x, y)$
for i in 1:x **do**
    for j in 1:y **do**
        if mask(i, j) **and** (LSC$_{\bar{R}}(i, j) > 0.999$) **then**
            $\theta(i, j)$ =LSC$_{\bar{\theta}}(i, j)$
        end if
    end for
end for
p=sum(valid($\theta(x, y)$))

Both algorithms use a very high variance threshold, namely 0.999. This ensures that only orientation estimates that are very close to their neighbours estimates are inserted into the orientation field. Figure 6.8 illustrates the result of orientation field growing,

where the majority of the remaining missing values in (a) and (b) are either: *minutiae points* or the centre of *Level I structures*.



(a) Delta (b) Loop

(c) Loop with creases (d) Vestige with creases

Figure 6.8.: Growing orientation estimates

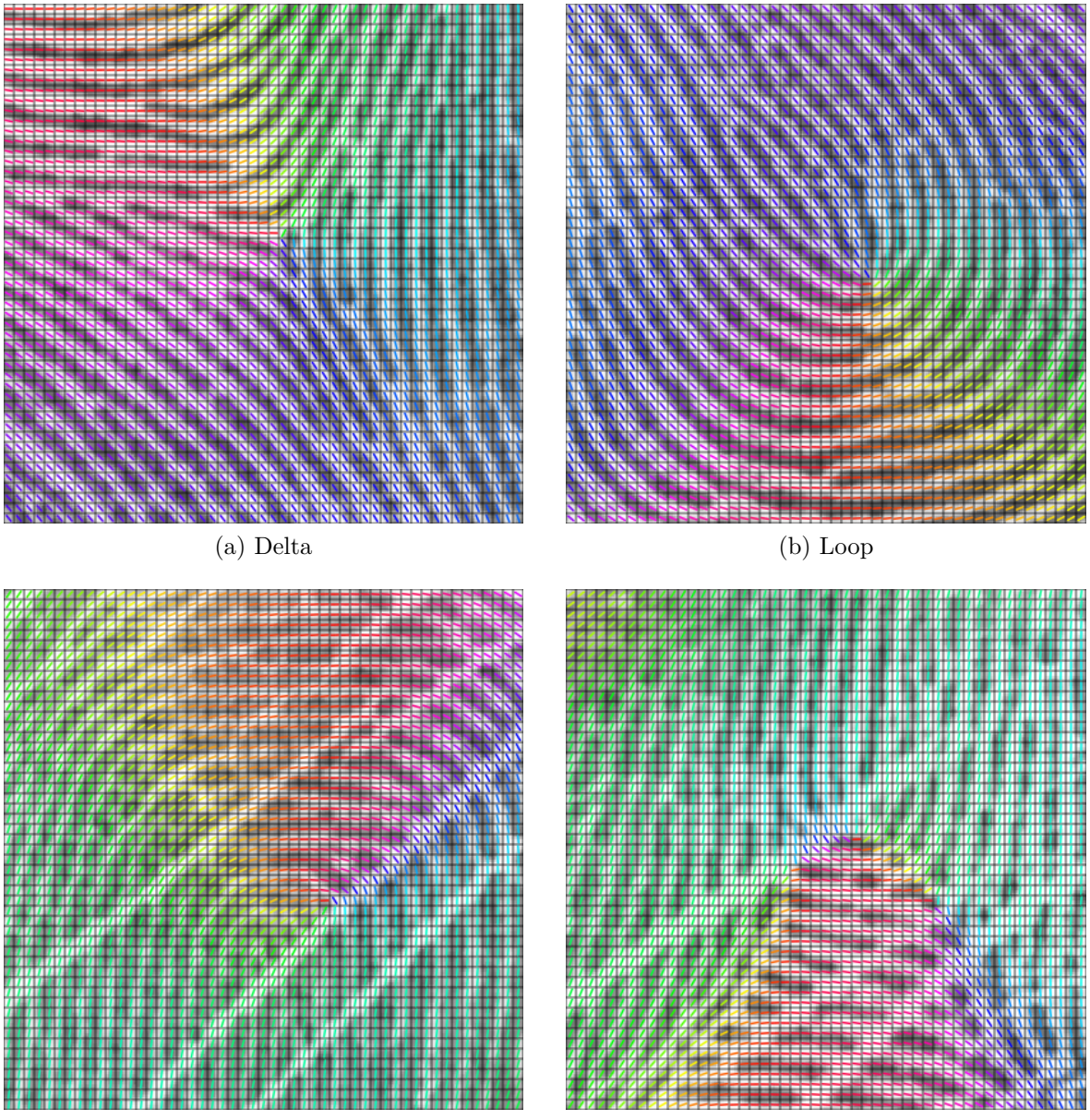
Figure 6.8 (c) and (d) illustrate the effectiveness of the adaptive local neighbourhood filtering method. The majority of the artefacts in the orientation field, due to creases and minutiae, have been suppressed and where the effects of creases still the remain, the orientations are comparable to the surrounding the friction ridges.

Step 7, 8 and 9

The circular mean, in which the individual pixels are weighted according to their variance, is applied iteratively to the orientation estimates $\hat{\theta}$, using LCS where $r = 3$ and $n = 1$. This has two effects upon the orientation field. Firstly, it reduces the affect of any large local deviations. Secondly, it fills in any remaining gaps within orientation

field.

Figure 6.9 illustrates the result of applying the entire algorithm ALNF to a raw orientation field with magnitude removal.



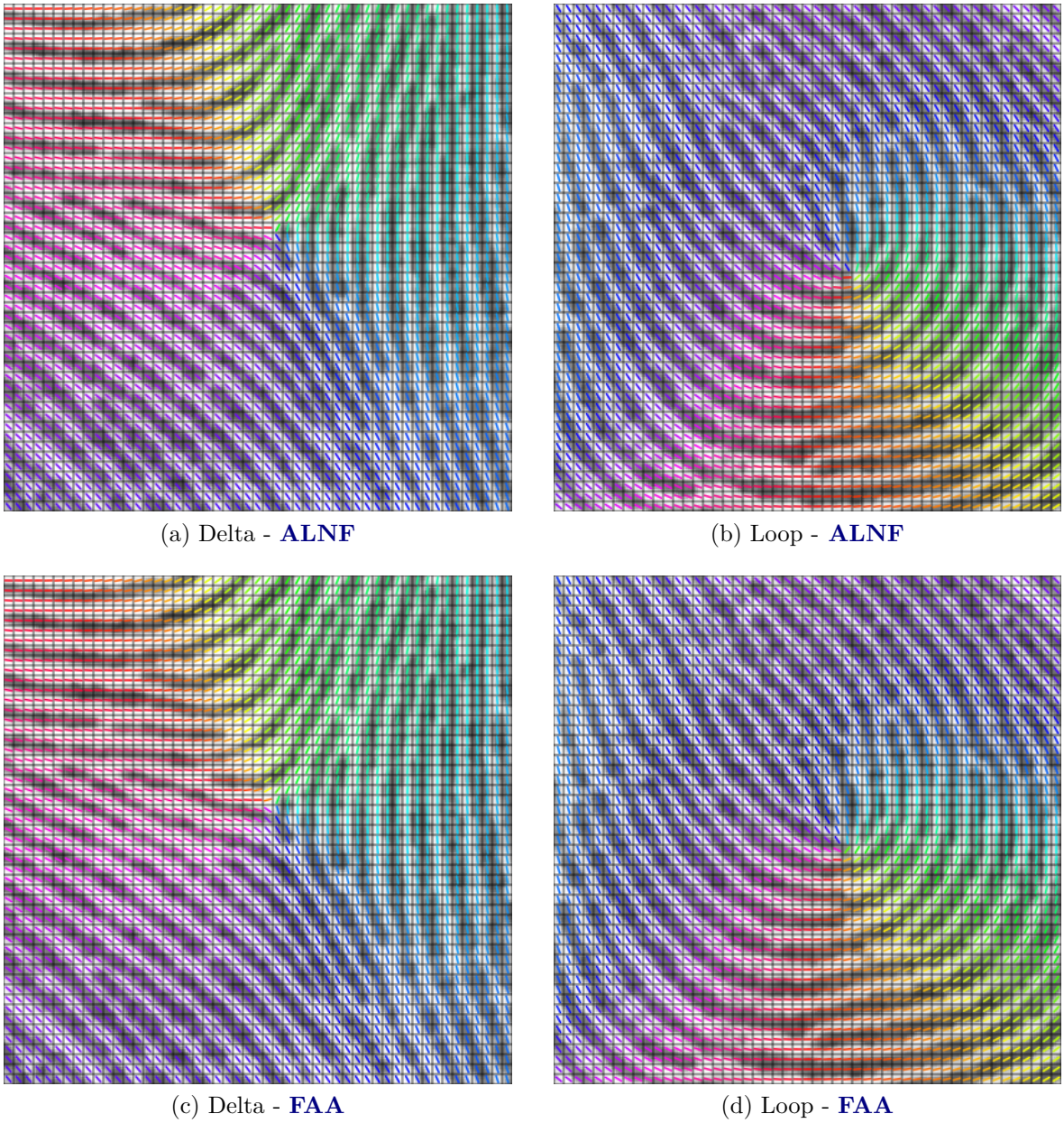
(a) Delta (b) Loop

(c) Loop with creases (d) Vestige with creases

Figure 6.9.: Adaptive Local Neighbourhood filtering on parts of a palmprint

6.4. Comparison Between Adaptive Local Neighbourhood Filtering and a Current Technique

Figures 6.10 and 6.11 illustrate a comparison of ALNF with the filtered orientation estimation method employed in the APIS [27] (FAA). The differences in figure 6.10 are relatively subtle, however the orientation estimates generated by ALNF around the centre of both the loop and delta structures, appear slightly more accurate.



(a) Delta - **ALNF**

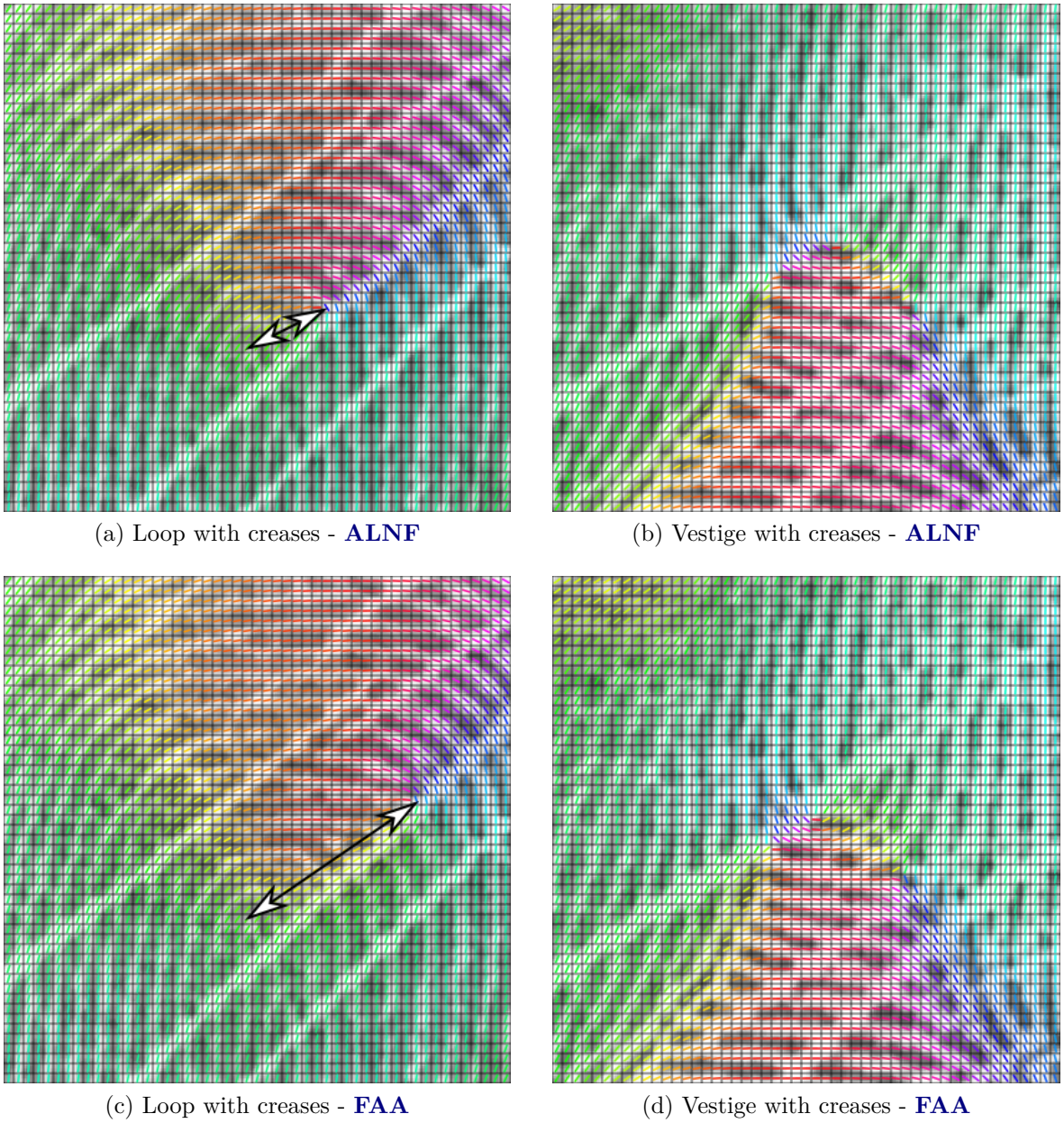
(b) Loop - **ALNF**

(c) Delta - **FAA**

(d) Loop - **FAA**

Figure 6.10.: Comparison of orientation estimates of **ALNF** and **FAA**, on two typical Level I structures

The differences, in the presence of creases, are much more obvious in figure 6.11. For both the loop and the vestige, the ALNF orientation estimates are more accurate than FAA. In figure 6.11 (a), the centre of the structure (illustrated by the arrow), generated by ALNF is shifted by approximately 3 friction ridges, whereas FAA, shifts the centre approximately 8 friction ridges from the correct position as shown in 6.11 (c), also illustrated by an arrow.



(a) Loop with creases - **ALNF**

(b) Vestige with creases - **ALNF**

(c) Loop with creases - **FAA**

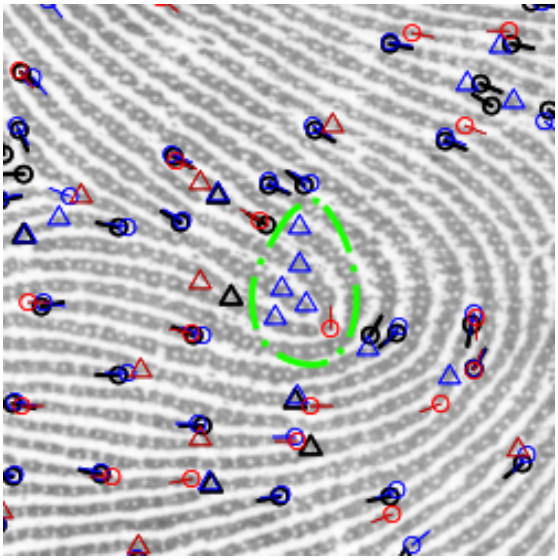
(d) Vestige with creases - **FAA**

Figure 6.11.: Comparison of orientation estimates of **ALNF** and **FAA**, on two Level I structures which have creases running through their friction ridges

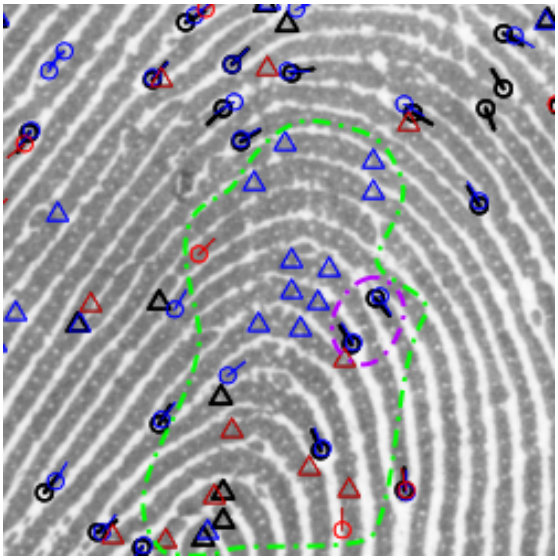
6.4.1. Minutiae Comparison

As illustrated in figure 6.11, it is difficult to see the slight variations which arise for different orientation methods and even more difficult to predict their effect on the final classification.

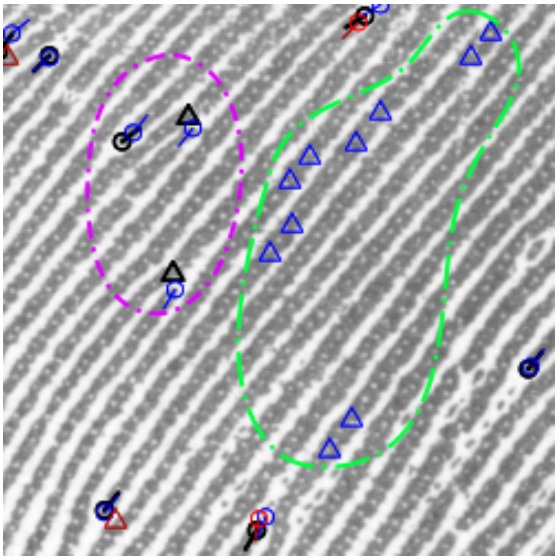
This section attempts to show the potential improvement in classification accuracy using ALNF, by replacing the orientation estimation stages of a minutiae based APIS (i.e. FAA [27]) with ALNF and noting the improved positional accuracy of the minutiae. These are then compared with the results from [39], denoted LPM (Latent Palmprint Matching), which used a frequency based approach to orientation estimation (section 3.2.2.2). Figure 6.12 illustrates the extracted minutiae for ALNF, FFA [27] and LPM [39].



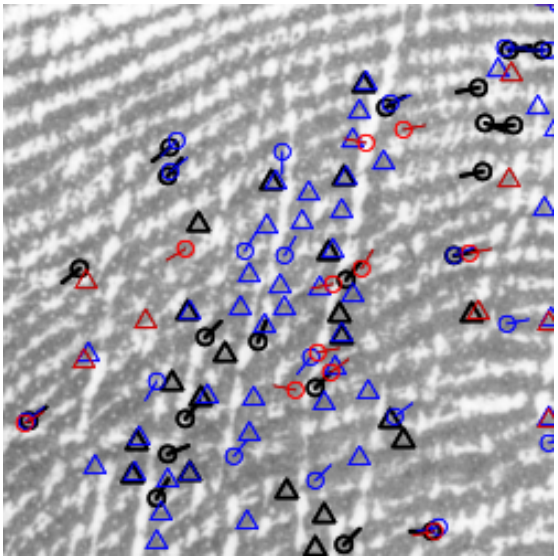
(a) A loop, false positive minutiae shown in the green region



(b) A loop, where false positive minutiae occur inside the green region: 4 for ALNF, 11 for FAA and 8 for LPM



(c) Consistent friction ridges, false positive minutiae shown in the green region and missed minutiae in the purple region



(d) Friction ridges with cross hatching

Minutiae type		Technique
End	Bifurcation	
○	△	**ALNF**
○	△	[27] **FAA**
○	△	[39] **LPM**

Figure 6.12.: Comparison of minutiae extraction, using the techniques described in this section (ALNF) and two APIS [27] and [39]

In all three cases the enhanced image (i.e. the contextual filtered image) from each technique had their minutiae extracted using the techniques described in section 3.3.

The following observations can be made of figure 6.12, in which a variety of different

properties of friction ridges are shown.

Neither [27, 39] extract reliable minutiae around level I structures (which is stated in their articles). In figure 6.12 (a), inside the highlighted green area LPM extracts 5 false positive bifurcation points (blue) and FAA has extracted a false positive end point (red).

In the green region in figure 6.12 (b), the false positive minutiae are as follows: 4 for ALNF, 11 for FAA and 8 for LPM. However, inside the purple region ALNF and FAA extracted very similar minutiae and LPM missed them. In both 6.12 (a) and (b) ALNF has extracted the most reliable minutiae around the centre of each structure.

Figure 6.12 (c) illustrates a relatively consistent set of friction ridges where FAA extracted 10 false positive bifurcations (inside the green region) incorrectly. LPM [39] missed both bifurcations and end points in the purple region.

In all 3 cases 6.12 (a), (b) and (c), the classification and position of minutiae using ALNF appears more accurate than the methods of FAA and LPM.

In figure 6.12 (d), part of the thenar with cross hatching is illustrated. Accurate minutiae extraction is obviously a weakness of the FAA method around these areas, where many false positive minutiae occur. The article describing LPM states that less minutiae points in areas where cross hatching occurs, produces better matching results [39]. This is evident in their minutiae extraction in this area, however there are both false positive minutiae and minutiae whose angle is inaccurate. On the other hand ANLF extracts a cluster of correctly extraced acted from ALNF in the top right hand corner.

The following chapter derives two distinct forms of features from the orientation estimate.

Chapter 7

Feature Extraction

7.1. Introduction

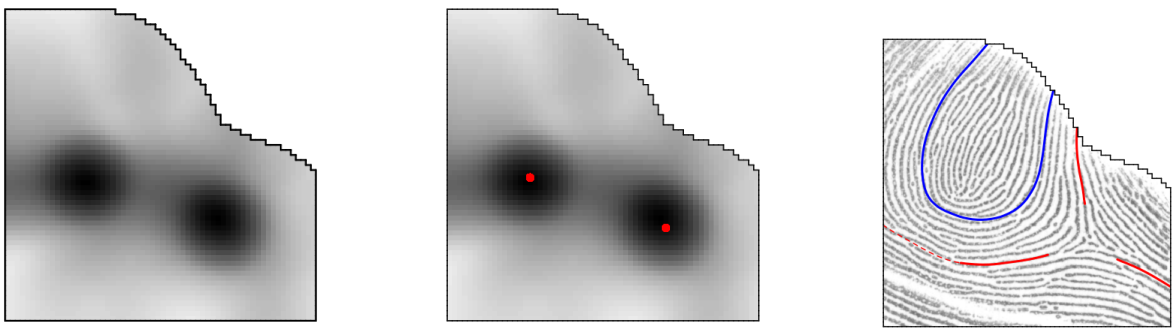
This chapter derives two distinct forms of feature from the orientation estimate derived in the previous chapter. The first, extracts and then characterises Level I structures using the statistics of a circular sweep about the centre of the structure. The second section creates two hierarchical tree-like representations of the homogeneous regions of circular variance within the palm. The nodes within the tree contain features which characterise the regions and will be used in matching in chapter 8.

7.2. Level I Detail: Structures

7.2.1. Overview of the Method

Figure 7.1 illustrates an overview of the stages undertaken in identifying, classifying and extracting the characteristic features of Level I structures. There are three main stages to this procedure, each of which is described by the indicated subsection:

1. Calculate the circular variance of the orientation field using LCS, subsection 7.2.3
2. Location of the central position of potential Level I Structures within the circular variance, subsection 7.2.4
3. Classification of the Level I Structures and the extraction of their characteristic features, subsection 7.2.5



(a) Calculating the circular variance → (b) Locating the position of structures, indicated by the red dots → (c) Classifying structures and extracting their features; delta - red, loop - blue

Figure 7.1.: Overview of the stages Level I structure extraction

This method results in the classification of the Level I structures within palm, in which the following characteristic features are extracted,

1. The type of structure: delta, loop or whorl
2. The position of the central point of a structure.
3. Paths which describe the structure, which are dependent upon the type of structure.

In figure 7.1 (c) a delta is illustrated by the red lines which follow the flow of the friction ridges from the centre of the structure. The blue lines in figure 7.1 (c) illustrate the path which describe the loop. Both the sets of red and blue lines were created automatically by using the orientation estimates from the orientation field $\hat{\theta}(x, y)$ from section 6.

Prior to presenting the method for extracting Level I structures, a method to automatically trace the flow of the friction ridges is presented. The resulting paths calculated by the tracer will be used as features for the Level I structures.

7.2.2. Tracing the Flow of an Orientation Field

By using the orientation field $\hat{\theta}(x, y)$ from section 6, a path can be extracted which will follow the path made by a friction ridge. An analogy for this method is to imagine the orientation field as the flow of a river and the output of this method being the path which a stick would make after being thrown into the river. The method described in this section is an iterative method which uses trigonometry to collect the Cartesian position at intervals along the path of a trace.

Since the friction ridge orientation field is axial by nature (i.e $\hat{\theta}(x, y) \in [0, \pi)$), at any point on the orientation field, two paths can be taken which would travel in opposite directions, both of which would be in the direction of the orientation field. Figure 7.2 illustrates this concept with two arrows at a particular point, where the orientation field $\hat{\theta}(x, y)$ and an 8 by 8 grid are overlaid upon part of a palmprint. The arrows, both start from the same point which is in the middle of the seventh block in both the x and y directions, respectively.

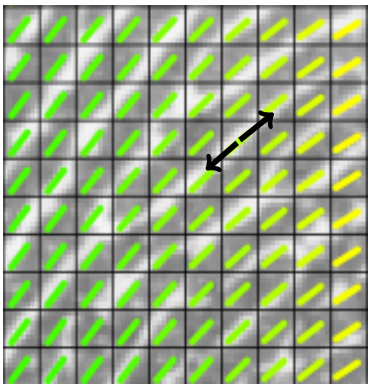


Figure 7.2.: Two possible directions of a path which follow the orientation field

Given that the orientation at the start point (7, 7) in figure 7.2 is $\hat{\theta}(7, 7) = 0.7037$ (to 4 d. p.), then the orientation field can be used to trace the direction of the flow of friction ridges, thus:

$$\begin{aligned} x_a &= 7 + \cos(0.7037) \\ &= 7.7625 \\ \\ y_a &= 7 + \sin(0.7037) \\ &= 7.6470 \end{aligned} \tag{7.1}$$

Or, since $\hat{\theta}(x, y)$ is axial data, the path could go in the opposite direction:

$$\begin{aligned} x_b &= 7 + \cos(0.7037 + \pi) \\ &= 6.2375 \\ \\ y_b &= 7 + \sin(0.7037 + \pi) \\ &= 6.3530 \end{aligned} \tag{7.2}$$

Where (x_a, y_a) and (x_b, y_b) are the positions which are a unit distance from the starting point.

The first solution (equation 7.1) is the end of the top arrow in figure 7.2 and the second (equation 7.2) is the end of the lower arrow in figure 7.2.

By choosing a starting position *a priori*, the trace can be forced to move in a specific direction. If it was desired that the trace should start off in a downward left direction, the trace could be *forced* in that direction. Let us pick a point in the opposite direction, (8, 8). Then given the start point $\hat{\theta}(7, 7) = 0.7037$, the second position of the trace will be:

$$(x, y) = \begin{cases} x = x_a & \text{if } \sqrt{(x_1 - 8)^2 + (y_1 - 8)^2} > \sqrt{(x_2 - 8)^2 + (y_2 - 8)^2} \\ y = y_a & \\ & \\ x = x_b & \text{else} \\ y = y_b & \end{cases} \tag{7.3}$$

Where $x_a = 7 + \cos(0.7037)$, $y_a = 7 + \sin(0.7037)$, $x_b = 7 + \cos(0.7037 + \pi)$ and $y_b = 7 + \sin(0.7037 + \pi)$.

So there are two positions from the result of equation 7.3, both of which are at the end of each of the arrows in figure 7.2.

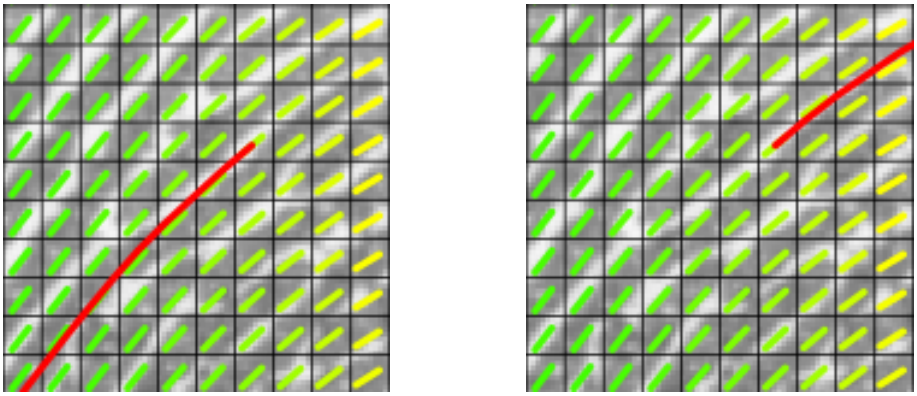
This method can be generalised so that a path of length n can be calculated. At each point along the path the decision of which choice to make is based upon being *forced* away from the previous point. This iterative method is possible due to the fact that the orientation of the friction ridges vary slowly in a local area; note it is rare that adjacent friction ridges are at an angle of $\frac{\pi}{2}$.

Given: a length n, an orientation field of friction ridges $\hat{\theta}(x, y)$, a starting point (x_1, y_1) and a point opposite to the desired direction of initial flow (x_0, y_0), the path of an orientation field can be found by algorithm 7.1.

The example shown earlier in this section (figure 7.2) is traced using algorithm 7.1 until it leaves the confines of the image; this is illustrated by the red lines in figure 7.3.

Algorithm 7.1 Tracer

Require: $\theta_{fr}(x, y)$, $x_n = x_1, x_2, ..., x_n$ and $y_n = y_1, y_2, ..., y_n$
Ensure: $x_n = x_1, x_2, ..., x_n$ and $y_n = y_1, y_2, ..., y_n$
1: **for** i in $1 : n$ **do**
2: $\theta_t = \theta_{fr}(x_n, y_n)$
3: $X_1 = x_n + \cos(\theta_t)$, $Y_1 = y_n + \sin(\theta_t)$
4: $X_2 = x_n + \cos(\theta_t)$, $Y_2 = y_n + \sin(\theta_t)$
5: **if** $\sqrt{(X_1 - x_{n-1})^2 + (Y_1 - y_{n-1})^2} < \sqrt{(X_2 - x_{n-1})^2 + (Y_2 - y_{n-1})^2}$ **then**
6: $x_{n+1} = X_1, y_{n+1} = Y_1$ **else** $x_{n+1} = X_2, y_{n+1} = Y_2$
7: **end if**
8: **end for**



(a) (b)

Figure 7.3.: Tracing the orientation field in opposite directions

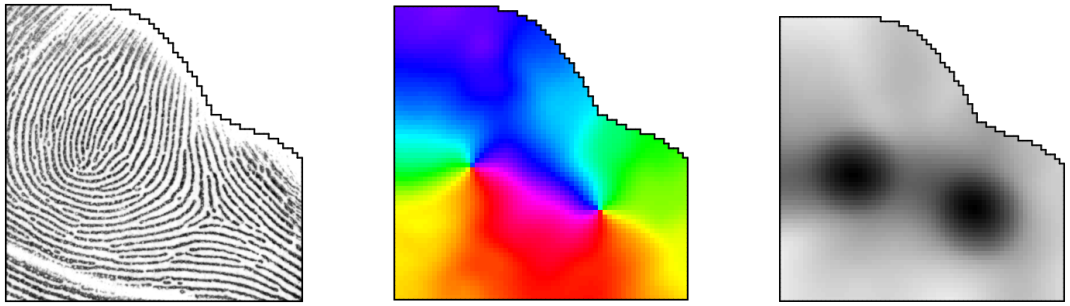
The output of the tracer described in this section has two purposes for the rest of this thesis. The tracer is used 1) to give a graphical representation of Level I structures (sections 7.2.5.3, 7.2.5.4 and 7.2.5.5) and 2) the paths are used to directly match Level I structures (section 8.2).

7.2.3. Circular Variance of the Orientation Field

Given the orientation field of the friction ridges ($\hat{\theta}(i, j)$ from section 6), the local circular variance $\theta_{1-\bar{R}}(i, j)$ is calculated using algorithm 6.1:

$$\theta_{1-\bar{R}}(i, j) = LCS(\hat{\theta}(i, j), n = 1, r = 10) \tag{7.4}$$

Figure 7.4 illustrates a graphical example of the process described above. Figure 7.4 (a) is part of an original palm, figure 7.4 (b) is the orientation of the friction ridges $\hat{\theta}(i, j)$ using the rainbow colour scheme and figure 7.4 (c) is the local circular variance of the orientation field of the friction ridges.



(a) Original palm

(b) Orientation of friction ridges $\theta_{fr}(i, j)$

(c) Circular variance $\theta_{\sigma^2}(i, j)$

Figure 7.4.: Calculating the circular variance $\theta_{\sigma^2}(i, j)$

The dark areas in figure 7.4 (c) represent areas in (a) and (b) where the orientation of the friction ridges varies more than other areas. Conversely, light areas in (c) represent areas in (a) and (b) where the orientation of the friction ridges is relatively similar.

By varying different values of r (the radius), the *smoothness* of the circular variance is altered. Smaller values of r produce unwanted peaks and preserve the position of the high variance regions in the circular variance, whilst larger values of r tend to *over* smooth the circular variance (shifting the position of structures). A value of $r = 10$ is robust enough to help identify the approximate position of structures, whilst smoothing over unwanted small deviations in ridge orientation.

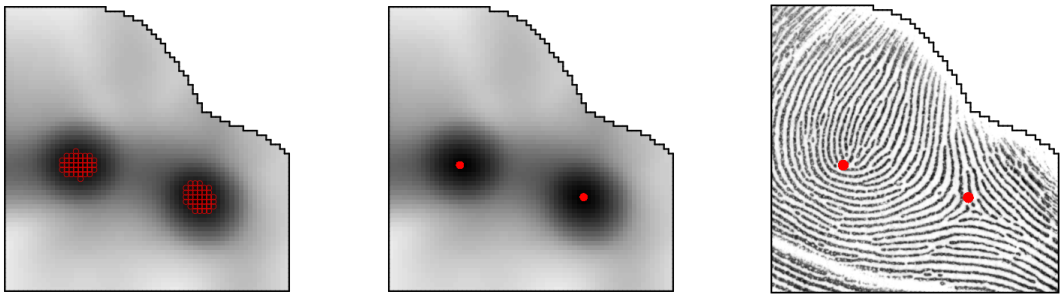
7.2.4. Locating the Position of Level I Structures

The approximate position of the centre point of Level I structures are where deep valleys occur in the circular variance $\theta_{1-\bar{R}}(i, j)$. The structures exist when the circular variance takes on a value greater than 0.9, this is illustrated in figure 7.5 (a) by the red circles.

Using a fixed threshold on $\theta_{1-\bar{R}}(i, j)$ a connected set of possible points is created, where each point has the potential to be the centre of a structure. Each connected set is potentially a structure and contains the centre point of the structure.

For each connected set, the point which has the maximum value in $\theta_{1-\bar{R}}(i, j)$, is illustrated in figure 7.5 (b). Figure 7.5 (c) illustrates this maximum point overlaid upon the original palmprint. Although this point is close to the centre of the structure, it is skewed towards the local area which has higher variance (i.e the delta (on the right in figure 7.5 (c)) is skewed above the centre of the structure).

To improve the position of the centre of the structure, the friction ridge orientation field is re-analysed, at each point, within each connected set as described in the next section.



(a) $\theta_{\sigma^2}(i, j) > 0.9$ Connected set

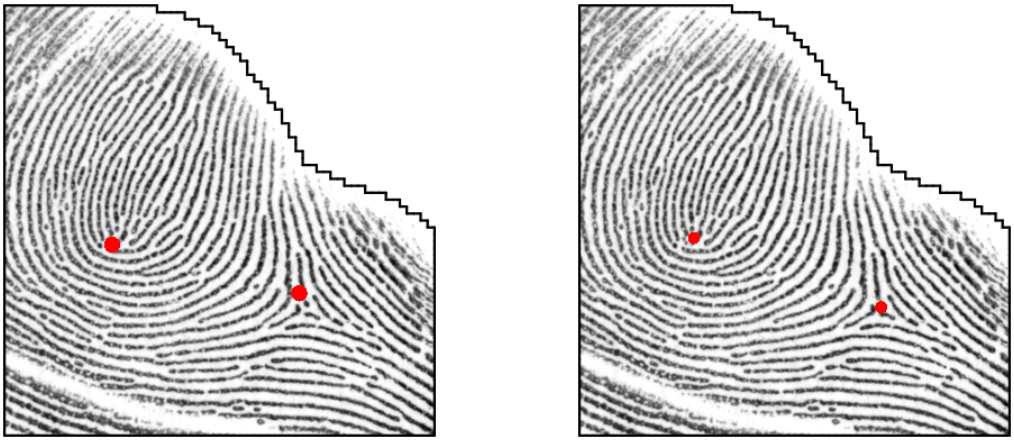
(b) Maximum point of the connected set overlaid upon θ_{fr}

(c) Maximum point of the connected set overlaid upon the palm

Figure 7.5.: Approximate position of the centre of Level 1 structures

7.2.4.1. Improving the Position of the Centre of a Structure

If the circular variance (equation 7.4) with a smaller radius ($r = 1$) is calculated, the result is a circular variance that is more sensitive to local deviations. Then, the maximum value for each structure (i.e the points illustrated in 7.5 (a) in a new circular variance for $r = 1$), provides an improved estimate for the centre of level I structures. The improvement in structure position is illustrated in figure 7.6.



(a) Original position

(b) Improved position

Figure 7.6.: Improved Structure position, where the centre of the structure is represented by the red dots

7.2.4.2. Summary

The central position of Level 1 structures are found by the following stages:

1. The circular variance of the friction ridges orientation is calculated with a *large* radius of $r = 10$.

2. The circular variance (from stage 1) is thresholded , which results in a connected set of possible points, each of which has the potential to be the central point of the structure.
3. The circular variance of the friction ridges orientation is calculated for the connected set with a *small* radius of $r = 1$.
4. For each connected set (stage 2), the centre of a structure is where the point from the connected set takes its maximum from the circular variance from stage 3.

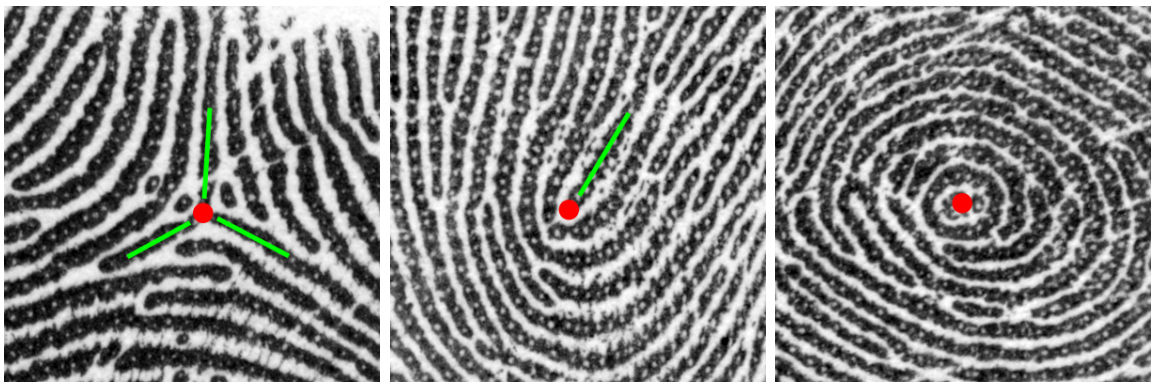
The position of each potential structure is now passed on to the classifying stage.

7.2.5. Classifying Level I Structures

7.2.5.1. Introduction

Before the method to classify Level I features is presented, a rationale for how to identify them is introduced. This rationale is based on the knowledge obtained from section 2.3.

Consider the orientation of the friction ridges, in the area close to the centre of a Level I structure, which are illustrated in figure 7.7. A red dot has been placed at the centre of each structure, and green lines have been placed where the friction ridges flow directly out, away from the centre of the structure.



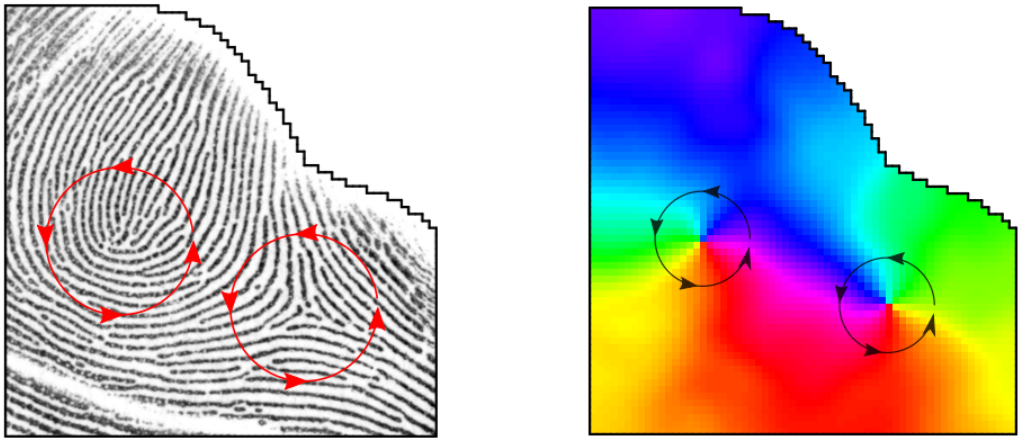
(a) Delta (b) Loop (c) Whorl

Figure 7.7.: The unique friction ridge flow which identifies each structures

Figure 7.7 (a) illustrates that there are *three* sets of friction ridges which flow directly out of a delta, *one* friction ridge flows directly out of a loop (figure 7.7 (b)) and there are *no* friction ridges which flow directly out of a whorl (figure 7.7 (c)).

7.2.5.2. Method for Classifying Level I Structures

By using the rationale above, structures can be classified by the number of friction ridges that flow directly out from the centre of the structure. By performing a full sweep (2π) about the centre of a structure, the orientation estimates can be collected and analysed. This circular sweep starts from the three o'clock position and is illustrated by the red arrow in figure 7.8 (a) and (b), over the palmprint and the orientation field respectively.



(a) Palmprint (b) Orientation field

Figure 7.8.: Circular sweep around the centre of a potential structure

The sweep is defined as a list $\theta_{2\pi}(n)$ and has a length of $n = 128$. So, $\theta_{2\pi}(n)$ is a set of axial data which represents the values of the orientation field collected in a circular sweep starting at the three o'clock point, and sweeping anti clockwise, a full 2π back to its starting position. Given a structure's centre point (x, y), the coordinates of the circular sweep are:

$$\theta_{2\pi x}(n) = x + r_1 \cos\left(\frac{2\pi n}{128}\right) \tag{7.5}$$

$$\theta_{2\pi y}(n) = y + r_1 \sin\left(\frac{2\pi n}{128}\right) \tag{7.6}$$

Where $r_1 = 4$ is the radius of the sweep, $\theta_{2\pi x}$ and $\theta_{2\pi y}$ are the coordinates in the x and y directions respectively.

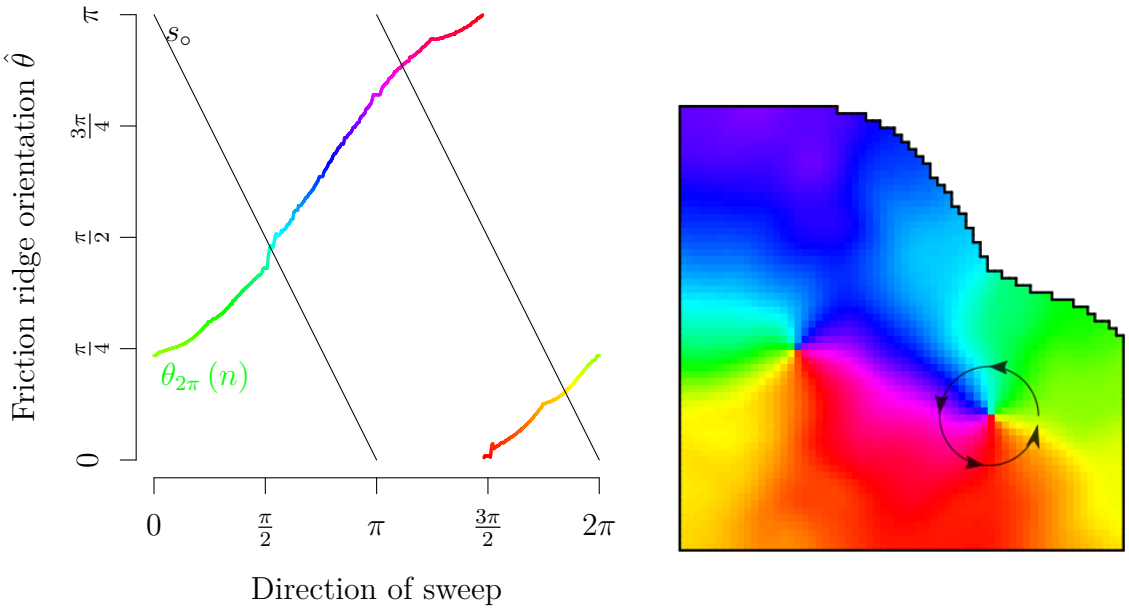
The orientation estimates about each of the structures are now collected from the orientation field $\theta_{2\pi}(n) = \hat{\theta}(\theta_{2\pi x}(n), \theta_{2\pi x}(n))$.

By using $\theta_{2\pi}(n)$ on the partial palmprint illustrated in figure 7.8, the following two sections describe and illustrate the classification method for deltas, loops and whorls.

7.2.5.3. Deltas

It has been shown that deltas have three perpendicular friction ridges flowing from the centre of a structure, section 7.2.5.1. Figure 7.9 (a) illustrates $\theta_{2\pi}(n)$ from the structure from figure 7.9 (b). The two lines represent:

- A plot of $\theta_{2\pi}(n)$ (in a rainbow colour map, in the y direction), against the direction of the circular sweep (x direction).
- A plot of the orientation of the circular sweep in the x direction (in black) and in the y direction, as perpendicular, axial data, which we denote $s_\circ$.



(a) Orientation Sweep (b) Orientation Field

Figure 7.9.: Classifying deltas by the circular sweep around the centre of a structure

The points at which the two lines ($\theta_{2\pi}(n)$ and $s_\circ$) intersect represent the points where the friction ridges are parallel to the orientation of the sweep. Points of intersection are where the friction ridge orientation flows directly from the centre of the structure. If $s_\circ$ (black line in figure 7.9 (a)) is subtracted from the orientations collected from the sweep (i.e $\theta_{2\pi}(n)$), a measure of the distance is obtained, which is the directional angular distance in Radians of which the friction ridge deviates from the circular sweep, which we denote as fr_θ.

$$fr_\theta(n) = dad(\theta_{2\pi} - s_\circ) \tag{7.7}$$

Figure 7.10 illustrates a plot of $fr_\theta(n)$, where the plot crosses zero at three distinct points.

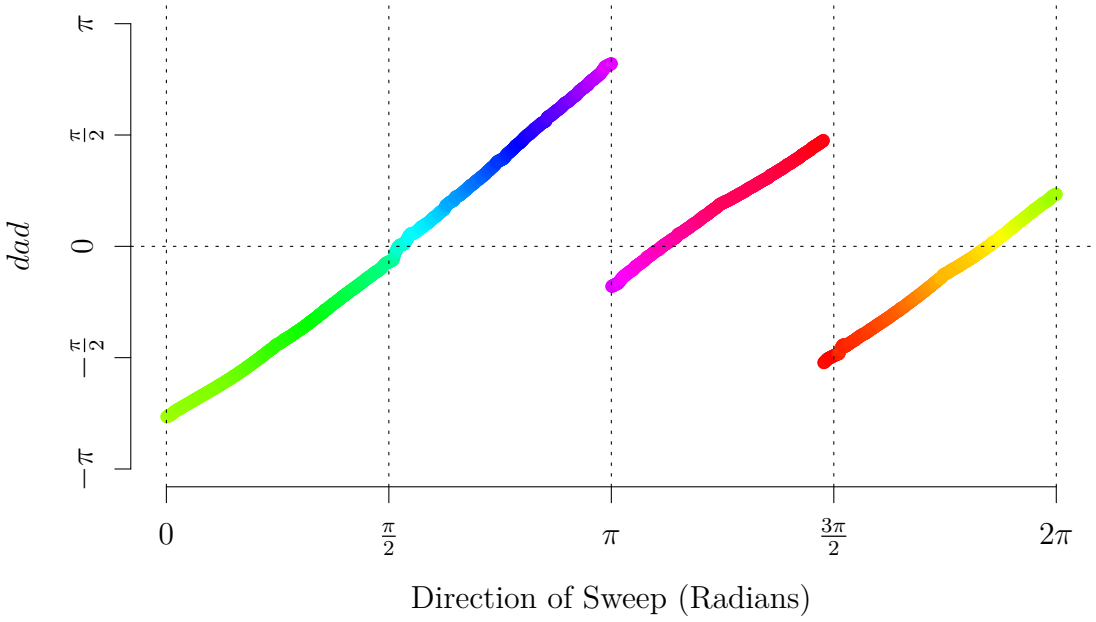


Figure 7.10.: The directional angular distance in which the friction ridges deviate from the circular sweep. The zero crossing show where $\theta_{2\pi}$ and $s_\circ$ are parallel.

Any structure whose $fr_\theta(n)$ crosses zero at three distinct points is now classified as a delta. Furthermore, the points at which $fr_\theta(n)$ crosses zero, show where $\theta_{2\pi}$ and $s_\circ$ are parallel. Which are the angles at which the friction ridges are flow directly out from the centre of the structure, which are denoted $\curlywedge_{i,\theta}$ where $i = 1, 2, 3$. Figure 7.10 illustrates that the angles in this example are $\curlywedge_{1,\theta} = 1.6352$, $\curlywedge_{2,\theta} = 3.4975$ and $\curlywedge_{3,\theta} = 5.7984$ in Radians to 4 d.p. respectively. The centre point of the delta is denoted $\curlywedge_{\oplus} = (x,\, y)$.

The structure is now classified as a delta and the paths of the three friction ridges which flow directly out of a delta (zero point crossing from equation 7.7) are extracted using the tracer from section 7.2.2.

The tracer is run three times, which results in a set of coordinates describing the path of each of the three friction ridges and are denoted as $\curlywedge_{i,n} = (x_{i,n}, y_{i,n})$, where $i = 1, 2, 3$ (for each path) and n is the length of each path. The position of the starting point (i.e $n = 1$) for each path is calculated by:

$$\curlywedge_{i,1} = \left(\curlywedge_{\oplus,x} + 2\cos\left(\curlywedge_{i,\theta}\right),\ \curlywedge_{\oplus,y} + 2\sin\left(\curlywedge_{i,\theta}\right)\right) \tag{7.8}$$

Where $\curlywedge_{\oplus,x}$ and $\curlywedge_{\oplus,y}$ are the coordinates of the centre point of the structure.

The tracer can now be run for each of the three friction ridges which flow directly out of a delta, from the starting points $\curlywedge_{i,1}$ and forced away from the centre point $\curlywedge_{\oplus}$.

The tracer is run until the path encounters the boundary of $\hat{\theta}$; at which point the tracer stops. The path of each trace is stored and used directly in a matching procedure.

Figure 7.11 illustrates the paths (in red) that each particular friction ridge takes. The thick red line illustrates the path from the starting position (i.e zero point crossing from fr_θ) to either 1) a length of 60 pixels or 2) the boundary of the valid palm. The

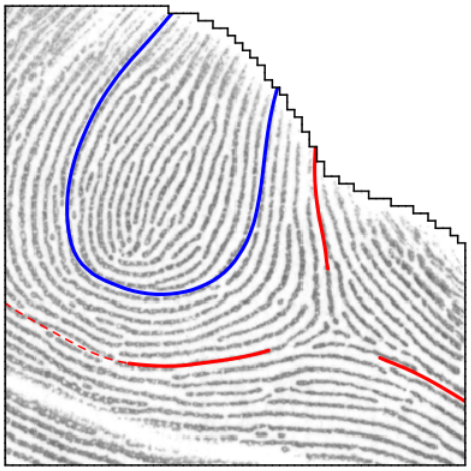


Figure 7.11.: The paths of delta (red) which have been automatically extracted. The red lines represents the three paths of the friction ridges up to 60 pixels from the starting point, the dotted line is greater than 60 pixels.

dotted red line illustrates the path to the boundary that is over 60 pixels from the starting point.

The attributes stored for each delta are: 1) The structure is classified as a delta $\curlywedge$ 2) The structures centre point $\curlywedge_{\oplus}$ 3) The three angles at which the friction ridges flow directly out of the centre of the structure $\curlywedge_{i,\theta}$ 4) The three paths of the friction ridges $\curlywedge_{i,n}$. This can be expressed as the set:

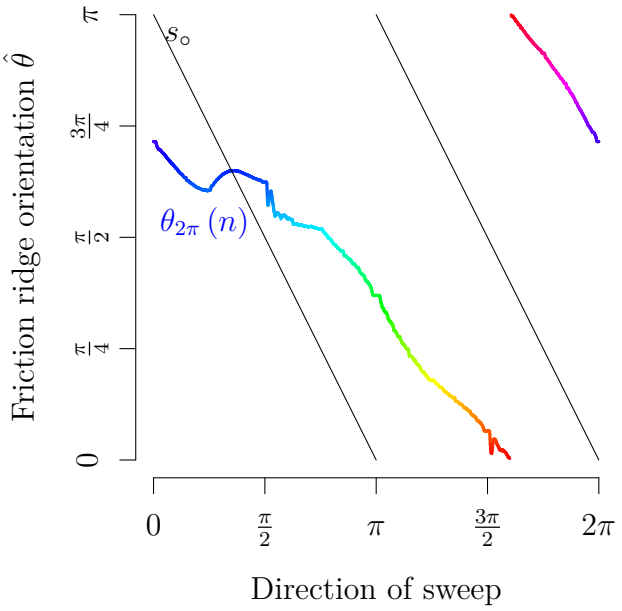
$$\text{Structure} = \begin{bmatrix} \curlywedge & \text{Type} \\ \curlywedge_{\oplus} & \text{Centre point} \\ \curlywedge_{i,\theta} & \text{Orientation of friction ridges} \\ \curlywedge_{i,n} & \text{Path of friction ridges} \end{bmatrix} \tag{7.9}$$

This set is used in the matching stage, which is described in section 8.2.2.

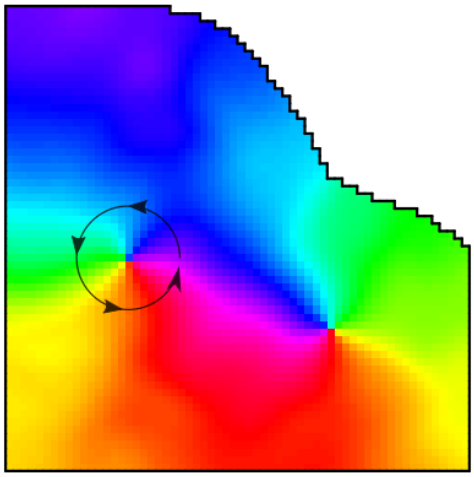
7.2.5.4. Loops

Loops have one friction ridge flowing directly from the centre of the structure, see section 7.2.5.1. Figure 7.12 (a) illustrates the orientation estimates from the circular sweep $\theta_{2\pi}(n)$ taken from the centre of the structure from figure 7.12 (b).

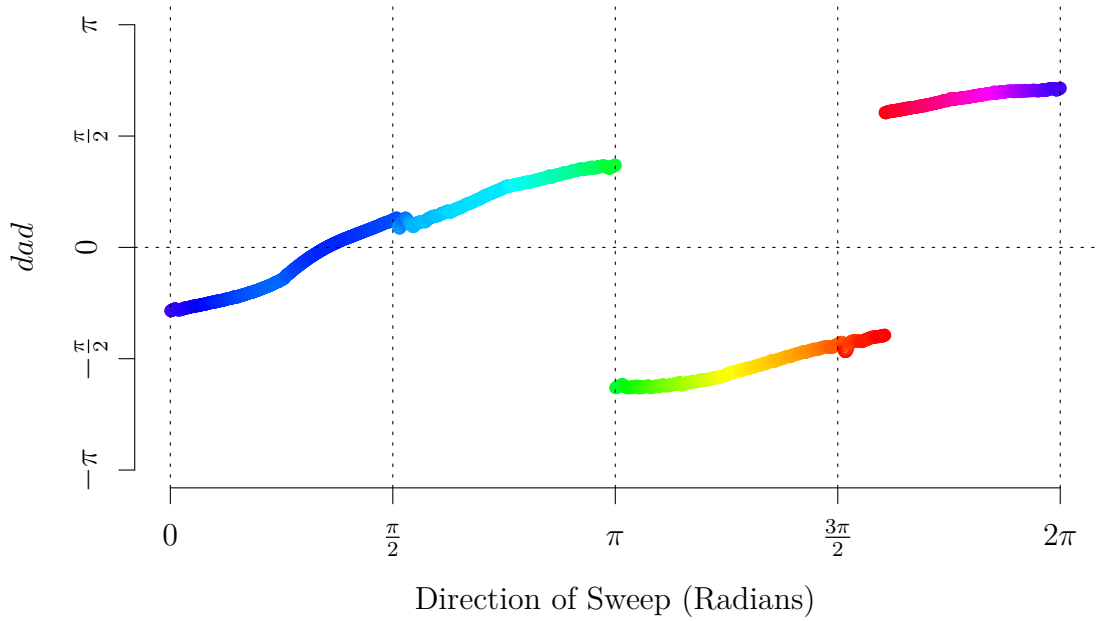
The same method is used for classifying loops as for classifying deltas, (i.e equation 7.7), except that loops have one zero crossing for fr_{θ}. Figure 7.12 (c) illustrates fr_{θ} for the example shown in (b), where the structure on the left is a loop. The friction ridge that flows into this loop is at an angle of $\Cap_{\theta} = 1.0983$ Radians to 4 d.p. and shall be denoted the *primary angle* of the structure.



(a) Orientation Sweep



(b) Orientation Field



(c) The directional angular distance in which the friction ridges deviate from the circular sweep. The zero crossing show where $\theta_{2\pi}$ and $s_\circ$ are parallel.

Figure 7.12.: Classifying loops, the zero crossing is the *primary angle* of the structure

For loops, the tracer is implemented in a different manner to that of deltas. Loops have one friction ridge flowing directly from the centre. Furthermore, it is not this friction ridge which identifies a loop, but the friction ridges opposite to this friction ridge (i.e the friction ridges which make the loop). In figure 7.13 the angle of the loop is $\circledcirc_\theta = 1.0983$, so the tracer is run on the opposite side from the friction ridge which is $\circledcirc_\theta + \pi = 4.23$.

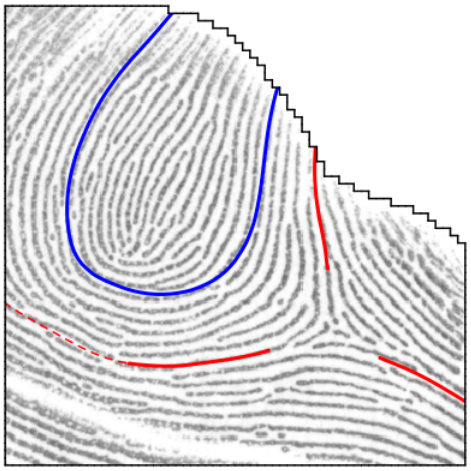


Figure 7.13.: The paths of loop (blue) which have been automatically extracted.

The centre point of a loop is denoted $\circledcirc_{\oplus} = (x, y)$. Then two tracers $\circledcirc_{i,n}$ (where $i = 1, 2$ and n is the length of each trace) are run in opposite directions, to form the structure around the centre of the loop. The starting point for the paths of the traces are:

$$\circledcirc_{i,1} = (\circledcirc_{\oplus,x} + 2\cos(\circledcirc_{\theta} + \pi), \circledcirc_{\oplus,y} + 2\sin(\circledcirc_{\theta} + \pi)) \tag{7.10}$$

Where $\circledcirc_{\oplus,x}$ and $\circledcirc_{\oplus,y}$ are the coordinates of the centre point of the structure.

Figure 7.13 illustrates the path the friction ridges of the loop takes (in blue), until it meets the boundary of $\hat{\theta}$. The physical structure of a loop is longer than the structure of deltas, so the length of the tracer is longer to reflect this. The length of the bold line is set to 80 pixels, then like the delta, a dotted line runs to the boundary.

The two paths that are stored and used directly for matching. So the attributes stored for each loop are: 1) The structure is classified as a loop $\circledcirc$ 2) The structures centre point $\circledcirc_{\oplus}$3) The orientation of the primary angle $\circledcirc_{\theta}$ 4) Two paths that trace the outline of the loop $\circledcirc_{i,n}$. This can be expressed as the set:

$$\text{Structure} = \begin{bmatrix} \circledcirc & \text{Type} \\ \circledcirc_{\oplus} & \text{Centre point} \\ \circledcirc_{\theta} & \text{Orientation of friction ridge} \\ \circledcirc_{i,n} & \text{Path of friction ridges} \end{bmatrix} \tag{7.11}$$

This set is used in the matching stage.

7.2.5.5. Whorls

Whorls are just a configuration of two loops. Once the structure classification stage (section 7.2.5.4) has been performed, if the centre points of two loops are within a distance of 200 pixels of each other (approximately 20 friction ridge widths), the two loops are potentially a whorl. If there are two loops within the given threshold distance, the loops are classified as loop 1 and loop 2 respectively and then the two paths made by each the loops are analysed. There are two criteria that have to be satisfied to classify a whorl:

1. If the traces from loop 1 converge within a distance of 18 pixels from the centre of loop 2 and
2. If the traces from loop 2 also converge within a distance of 18 pixels from the centre of loop 1

If the both criteria above are satisfied, the positions of both loops are stored, and they are classified as a whorl.

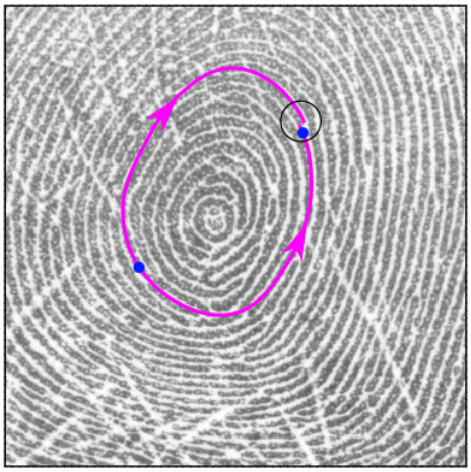


Figure 7.14.: Automatic tracing of a whorl, the two classified loops are illustrated in blue and the paths which converged the closest are illustrated in the black circle

The closest paths, which converge, are stored and used in the matching process.. Figure 7.14 illustrates a whorl in mauve and the paths that converge the closest, produced by the tracer.

The tracer is stopped at the point where it converges to the closest point, this is illustrated in figure 7.14.

So the attribute stored for each whorl are: 1) Classified as a whorl 2) The structures centre point 3) Two paths that trace the outline of the whorl.

7.2.5.6. Missed Classification

If a structure is not classified using the method described in sections 7.2.5.3 and 7.2.5.4, the classification is run again but this time using equations 7.5, 7.6 and 7.7, with a radius ($r_2 = 3$). If this results in a positive classification, then the Level I structure is classified accordingly. If the classification is positive, the tracer starts in the same position as the first attempt, otherwise, the trace would have an incorrect length.

7.2.6. Summary of Method

To summarise, the automatic Level I structure extraction technique presented in this section, results in the following four attributes:

1. Classification of the type of structure.
2. A central position for each structure.
3. The orientation of any friction ridges which flow from the centre of the structure.
4. A path for each of the friction ridges that describes the structure.

Since each structure contains different attributes, each structure has its own unique set of extracted features. The attributes of each structure are summarised in the table below.

Type of structure	No. p. ridges	Display Colour	No. of Paths
Loop	1	Blue	2
Delta	3	Red	3
Whorl	0	Mauve	2

Table 7.1.: Summary of structure attributes

7.3. Verification of Level I Structure Extraction

To test the efficacy of the automatic Level I structure extraction technique described in section 7.2, the algorithm was applied to 226 exemplar palmprints and the results manually verified.

Table 7.2 illustrates how the results of the verification was collected.

For each type of level I structure, the algorithm has three possible outcomes,

- **identified** - correctly classify a Level I structure indicated in green in table 7.2 or
- **missed** - not classify Level I structure indicated in red in table 7.2 or
- **false positive** falsely classify Level I structure (i.e. structure does not exist), indicated in blue in table 7.2

	Tri-radiate			Hypothenar			Thenar		
Palm	⅄	⋒	○	⅄	⋒	○	⅄	⋒	○
001	4 0 0	1 0 0	0 0 0	1 0 1	1 0 0	0 0 0	0 0 0	0 0 0	0 0 0
002	5 0 0	2 0 0	0 0 0	2 0 0	1 0 1	0 0 0	0 0 0	0 0 0	0 0 0
⋮									
226	3 1 0	2 0 0	0 0 0	1 0 0	0 0 0	0 0 0	1 0 0	1 1 0	0 0 0

Table 7.2.: Identified (green), missed (red) and false positive (blue) for Level I structures (⅄ is a delta, ⋒ is a loop and ○ is a whorl) in each region of each palm.

So the palm 001 in table 7.2 has in the tri-radiate: four deltas and a loop correctly identified, in the hypothenar: a delta and a loop correctly identified and a false positive delta and there are no structures in the thenar region.

By summing each of the the columns for each region, in table 7.2, it is possible to analyse the efficacy of the automatic Level I structure extraction algorithm for each region of the palm. These results are summarised in table 7.3. In the tri-radiate of all 226 palmprints there are 843 deltas, 242 loops and 1 whorl, of which 828 deltas, 222 loops and 0 whorls where correctly identified.

∑	Tri-radiate			Hypothenar			Thenar		
	⅄	⋒	○	⅄	⋒	○	⅄	⋒	○
Total	843	242	1	150	54	0	37	20	0
Identified	828	222	0	149	52	0	32	12	0
False positive	4	12	0	12	8	0	49	2	1

(a) The Sum of all Level I structure Classifications

%	Tri-radiate			Hypothenar			Thenar		
	⅄	⋒	○	⅄	⋒	○	⅄	⋒	○
Identified	98.22	91.74	0	99.33	96.3	0	86.49	60	0
False positive	0.48	5.13	0	7.41	12.9	0	60.49	14.29	1

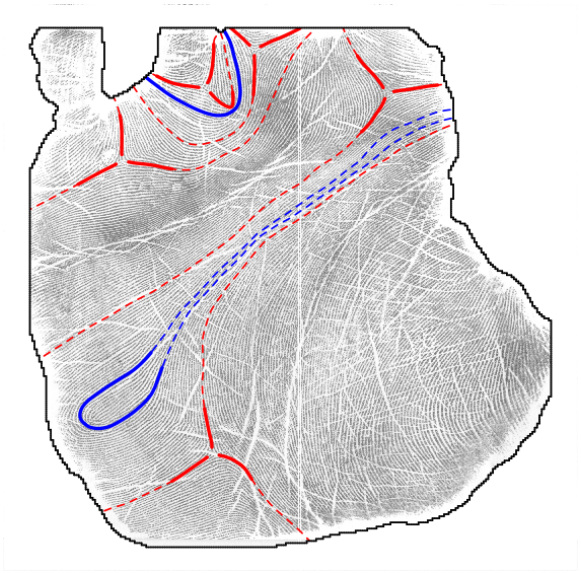
(b) Percentage of Level I structures

Table 7.3.: Summary of table 7.2, for Level I structures (⅄ is a delta, ⋒ is a loop and ○ is a whorl) in each region of each palm.

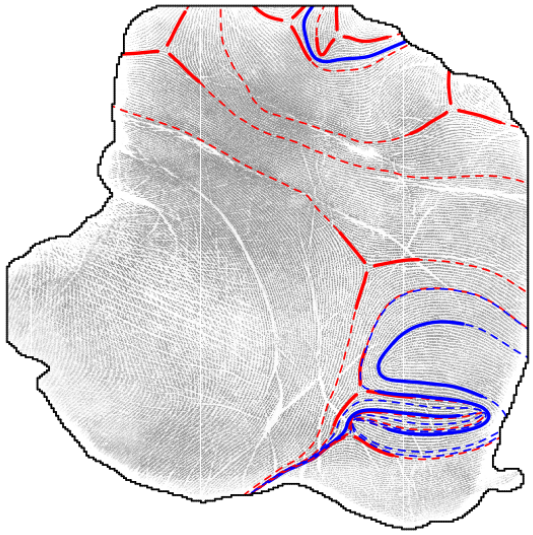
The results in table 7.3 show that the Level I structure extraction method works particularly well in the tri-radiate and hypothenar region of the palm.

Since the thenar region is dominated by creases (cross hatching see section 2.3.6.3), it is this region in which the adaptive local neighbourhood filtering method (see section 6.3) removes and replaces a relatively large number of orientation estimates. Therefore it is intuitive that the Level I structure extraction method is less accurate in this region.

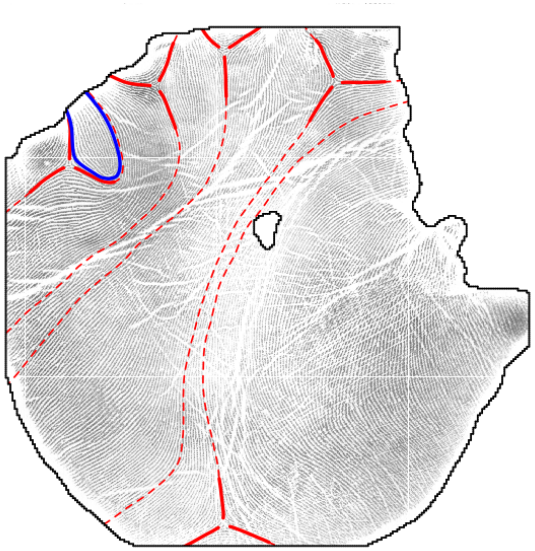
Examples of four different palms with their Level I structures correctly extracted are illustrated in figure 7.15, where deltas and loops are illustrated in red and blue respectively.



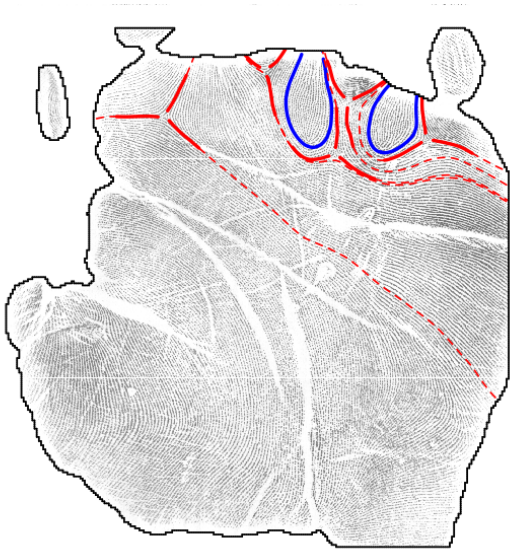
(a) Four deltas and a loop in the Triradiate, Carpel delta and loop in the hypothenar



(b) Four deltas and a loop in the Triradiate, Three deltas and three loops in the hypothenar



(c) Four deltas and a loop in the Triradiate, Carpel delta in the hypothenar



(d) Five deltas and two loops in the Triradiate

Figure 7.15.: Examples of correct Level I Structure extraction. Deltas are red and loops are blue.

The automatic identification of Level I structures is also a major step towards the automatic segmentation of an exemplar palm into its regions. The Tri-radiate region of every palm contains at least 3 deltas and normally contains loops, sometimes as many as four. Automatically classifying the structures in this region and using the direction that friction ridges flow out from the centre of the structure could lead to automatic segmentation of the three major regions.

7.4. Regions of Homogeneous Variance from Friction Ridge Orientation

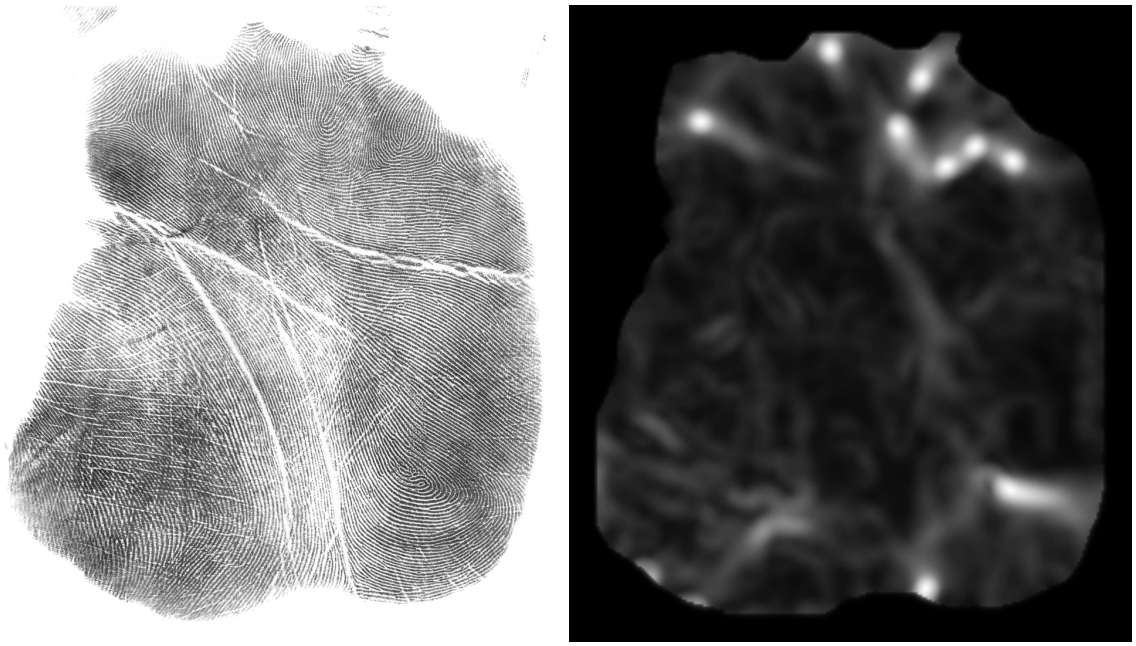
7.4.1. Introduction

This section describes and illustrates a novel technique that segments a palm into regions based upon the circular variance of the friction ridge flow. The results of the segmentation are represented by a hierarchical tree structure used in matching.

Section 6 describes how to calculate the orientation estimates of the friction ridge flow $\hat{\theta}(i, j)$ from a palmprint using the algorithm 6.2. The local circular variance of the orientation of friction ridge flow $1 - \bar{R}(i, j)$ is then calculated using algorithm 6.1. These values of the circular variance are used to segment the palm into regions of homogeneous variance of the friction ridge flow.

Various statistics for each of the segmented regions are then calculated and used in matching procedures, which are described in section 8.3.

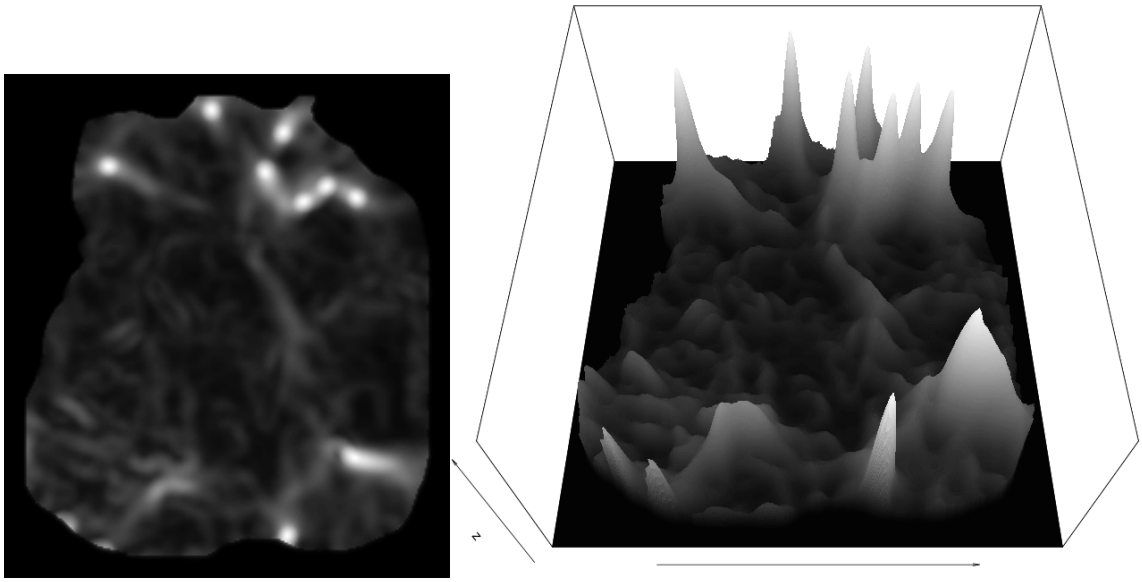
Consider the palmprint in figure 7.16 (a) and its corresponding local circular variance in figure 7.16 (b). The local circular variance has values between 0 and 1 and is illustrated in figure 7.16 (b) by a greyscale colour map, where 0 and 1 are represented by black and white respectively.



(a) Palmprint (b) Local circular variance $1 - \bar{R}(i, j) \in [0, 1]$

Figure 7.16.: A palmprint and its local circular variance

The local circular variance can be considered as a surface where the value of the local circular variance is the height of the surface at each particular point (i.e $z = 1 - \bar{R}(i, j)$); a visual representation of this is illustrated in figure 7.17.



(a) An image (b) A surface

Figure 7.17.: Different visual representations of $1 - \bar{R}\,(i,\, j) \in [0,\, 1]$

By applying strict inequalities on the surface z, connected homogeneous regions can be identified in local circular variance. Figure 7.18 illustrates the contours of the regions created by the five strict inequalities $1 - \bar{R} \geq [0.1,\, 0.3,\, 0.5, 0.7,\, 0.9]$. Sets of such contours are also known as *level sets*.

Level set methods have been widely used as a general framework in image processing, to segment images [87]. Segmenting images using level set methods, generally involves normalising or thresholding the entropy at each level [88].

Building upon methods for level set segmentation, techniques have been developed that segment images using sets of contours. Techniques such as the Waterfall transform [89], P-transform [90] and [91] use sets of contours as the basis for segmentation. For this application, level sets will be applied to the circular variance, rather than to the grey scale intensity.

7.4.1.1. What do such regions represent?

The algorithm in section 7.2 postulates that level I structures exits when $1 - \bar{R} > 0.9$, which is illustrated by the orange circles adorning the peaks shown in figure 7.18. From figure 7.18, it now can be seen that the orange regions bound the structures and are themselves sub levels contained within further sub levels, which culminate in the entire palmar surface.

From inspection of lower sub levels (i.e the green regions) in figure 7.18 and the palm from which is derived in figure 7.19, the following observations can be made. The green regions are either, sub levels of taller features or areas on the palm ,where more subtle variations in the friction ridge orientation exist (i.e. a funnel).

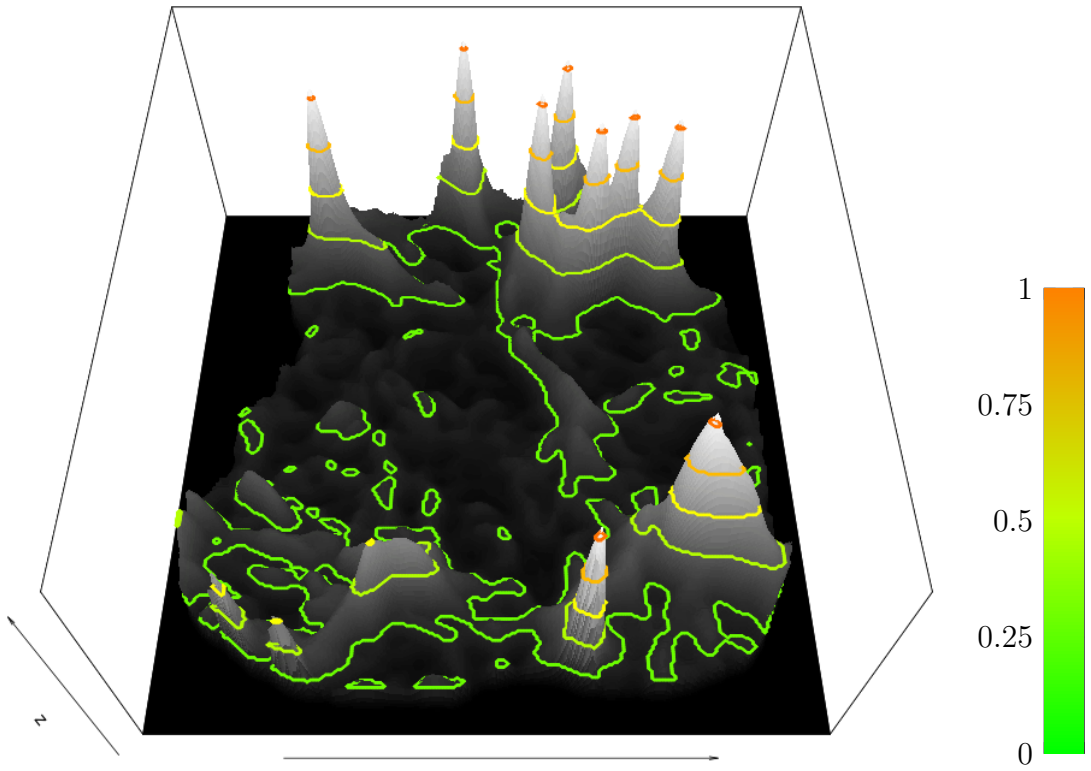


Figure 7.18.: Level sets of the circular variance at $1 - \bar{R} \leq [0.1, 0.3, 0.5, 0.7, 0.9]$

Figure 7.18 illustrates the level sets $1 - \bar{R} \geq [0.1, 0.3, 0.5, 0.7, 0.9]$ overlaid upon the palm in figure 7.19, in which the Level I structures have been identified and each of which is contained within a sub level of other regions.

Figure 7.19 also illustrates the subtle variations in the circular variance. Obviously, the peaks in the circular variance dominate the surface, but these smaller subtle variations can also be used to segment a palm's surface. The subtle variations exist in all of the major regions of the palm and correspond to some of the features used by forensic investigators. The funnel area in the hypothenar region has been identified in figure 7.19, which exists in the centre of one of the regions extracted using level sets $1 - \bar{R} \geq [0.1, 0.3, 0.5, 0.7, 0.9]$. The funnel area will be discussed further in section 7.4.5.

Figure 7.18 and 7.19 both illustrate the concept of level sets using only five levels in the circular variance. Since the subtle variations exist where the circular variance is below 0.2, many more levels than five are required, for the sets to be useful for matching purposes. To make best use of level sets for matching purposes, tests have shown that 100 level sets works well.

To summarise, using level sets at regular intervals on the circular variance, the palm can be segmented into regions. This method results in a contour map of the circular variance, that contains regions and sub regions (also known as sub levels for level sets) which has a hierarchical tree-like structure. It will be shown in the next section that by performing this region extraction method on both $1 - \bar{R}$ and $\bar{R}$ independently, two complementary tree structures can be extracted.

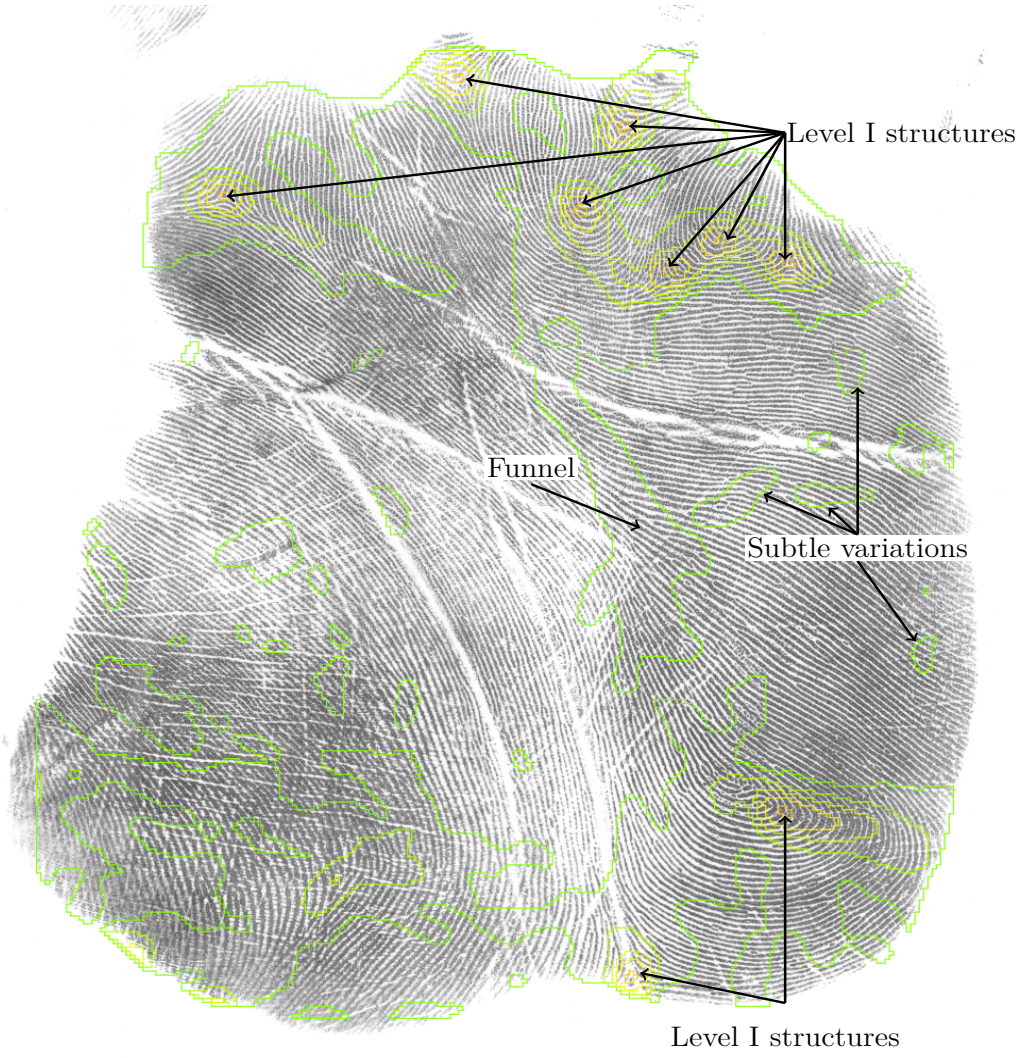


Figure 7.19.: Level sets of the circular variance at $1 - \bar{R} \geq [0.1, 0.3, 0.5, 0.7, 0.9]$ overlaid upon the palm

The following two sections describe and illustrate how to extract the sets and construct the trees.

7.4.2. Up Level Sets

To extract level sets from a circular variance image $1 - \bar{R}(i, j)$, a three dimensional array is created, denoted $U(i, j, n)$ where i and j are the number of rows and columns in $1-\bar{R}(i, j)$, respectively and $n = 100$. Then each interval n is defined by $1-\bar{R}(i, j) \geq \frac{n}{100}$. Therefore, for each n:

$$U(i, j, n) = 1 - \bar{R}(i, j) \geq \frac{n}{100} \tag{7.12}$$

So each n in $U(i, j, n)$, can be considered as a binary map which represent regions of the circular variance $1 - \bar{R}(i, j)$ for each interval.

A connected component algorithm [92] is now run over the first interval $U(i, j, 1)$ and any independent regions (i.e isolated 8-connected region) are labelled numerically. Any region which intersects the super set (i.e. the boundary), of the valid palmprint area

(i.e $U(i, j, 1)$), is eliminated. Therefore, an independent region from $U(i, j, 1)$ is a sub level of the boundary of $1 - \bar{R}(i, j)$. Thus an independent region in $U(i, j, 1)$ has the potential to contain further sub levels and so can be represented as node within a tree structure.

7.4.2.1. Building Level Set Trees

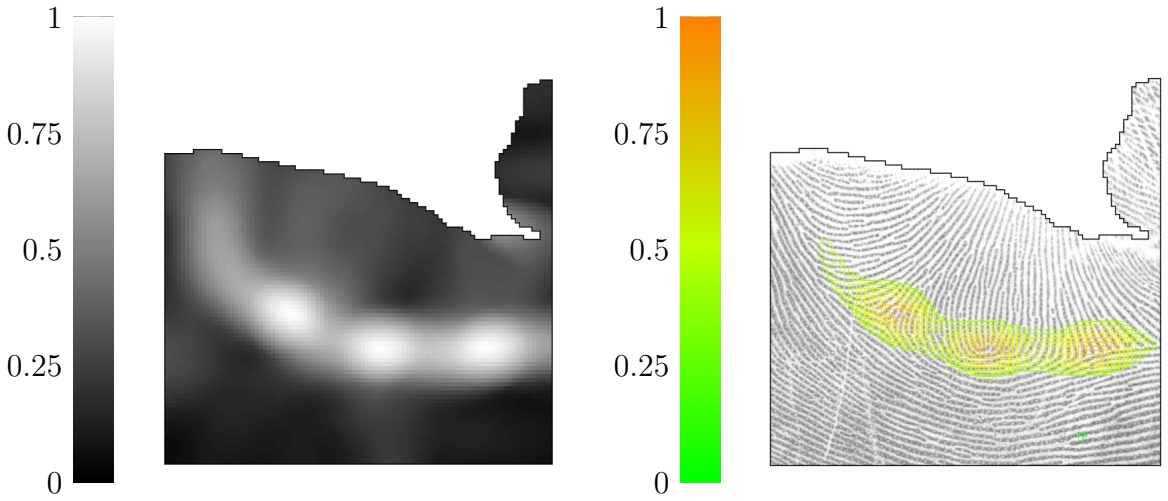
An array L is created, where each independent region in $U(i, j, 1)$ is a row entry containing its numerical label.

To create the tree, the following iterative process is run for each n such that $n \geq 2$:

1. A connected component algorithm [92] is now run over the level set $U(i, j, n)$ and any independent regions are labelled
2. Any region that intersects with the boundary is eliminated
3. If any region from $U(i, j, n)$ exists in the same position as a region from $U(i, j, n-1)$ (i.e. below it in the tree) then the corresponding region from $U(i, j, n)$ becomes a child node of $U(i, j, n-1)$.

Any region that intersects with the boundary is eliminated, because an incomplete region (i.e a region that intersects with the boundary) has an unknown area for which statistics cannot be derived.

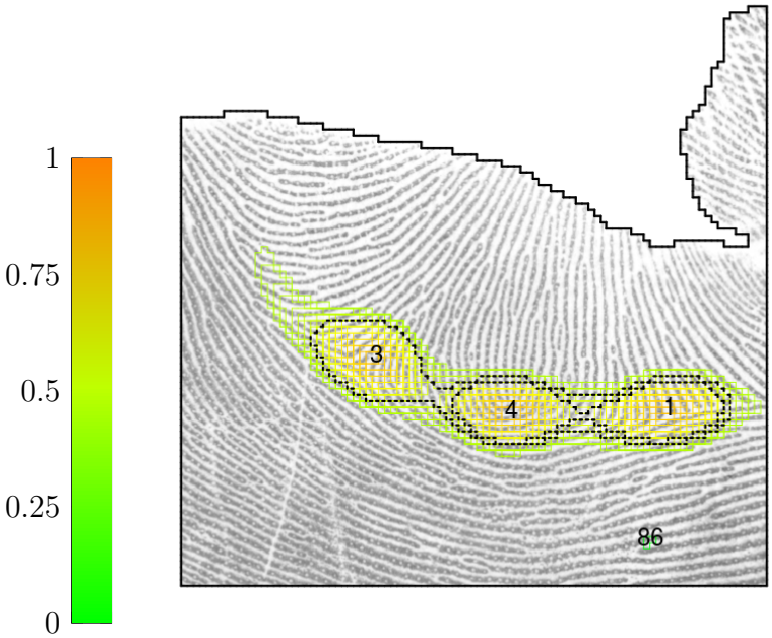
Figures 7.20 (a) and (b) illustrate the circular variance and the extracted regions overlaid upon the original palmprint. A green to orange colour map is used in figure 7.20 (b) to illustrate where the tree of enclosed regions are and where the range green to orange represents the height of the node in the tree.



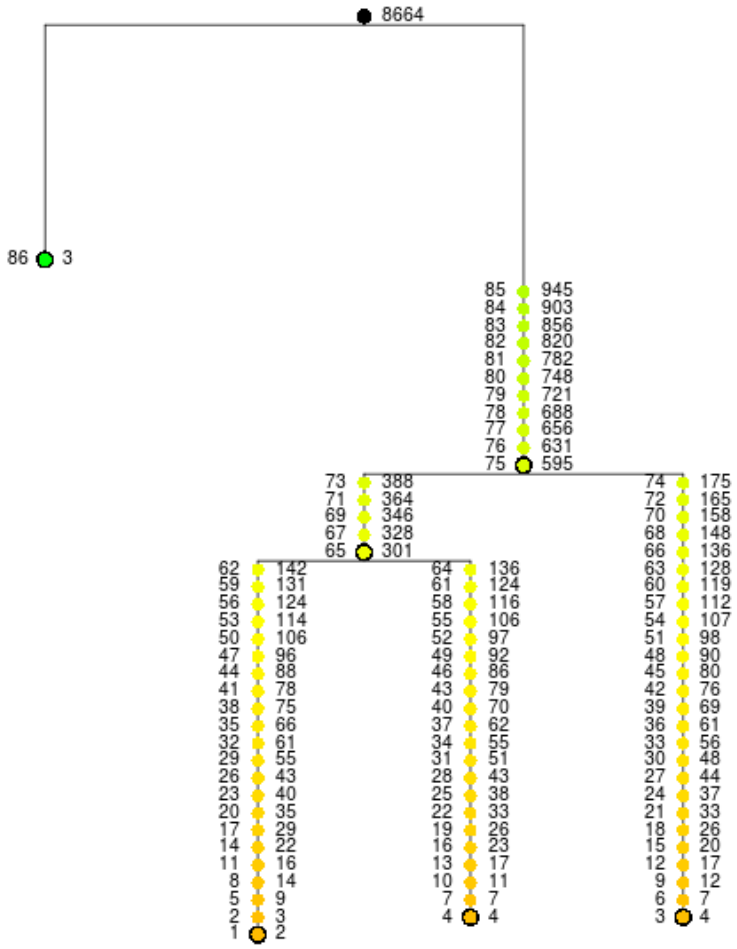
(a) Circular Variance (b) Contours on Palm

Figure 7.20.: Contours of the circular variance

Figure 7.21 (a) illustrates the regions of the circular variance and (b) illustrates these same regions represented as a tree.



(a) Contours on Palm



(b) Tree

Figure 7.21.: Contours of the circular variance and Corresponding Tree

Figure 7.21 (b) is the tree structure, where each node has:

- The region number on the left.
- The region size on the right.
- The colour of the node is representative of the level the contour at which it exists in the circular variance.

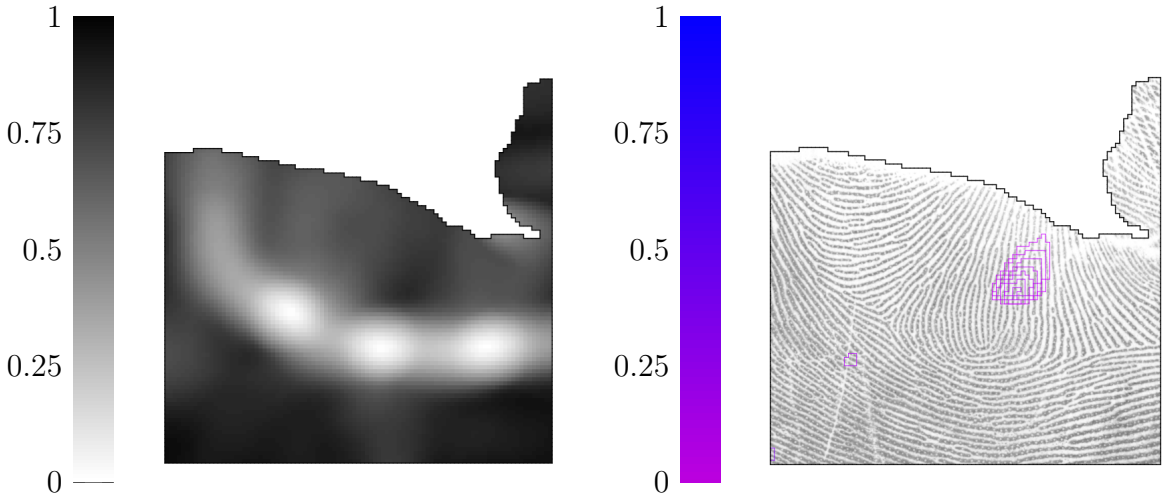
A node is outlined in black if either:

- The node is a leaf.
- The node is a parent of two or more branch structures.

The leaf nodes have been labelled in figure 7.21 (a), as have the parents of two or more branch structures. The trunk (or root) node in 7.21 (b) represents the valid palmprint area contained within the boundary, which was segmented by the algorithm 6.2 (ALNF) in chapter 5.

7.4.3. Down Level Sets

When creating the down level sets, the same method is used as the up level sets, except $\overline{R}(i, j)$ is used instead of $1 - \overline{R}(i, j)$ and the notation used is $D(i, j, n)$ instead of $U(i, j, n)$. So instead of extracting level sets starting in the base of a valley, we have flipped the circular variance, so theoretically, slices are being removed from the tops of peaks.



(a) $\overline{R}$ (b) Contours on Palm

Figure 7.22.: Contours of the circular variance

Figure 7.22 (a) illustrates $\overline{R}(i, j)$ and figure 7.22 illustrates the corresponding down level sets overlaid upon the palmprint. A purple to blue colour map is to illustrate the down level sets in palmprint.

7.4.4. Combined Up and Down Level Sets

Figure 7.23 illustrates both the up and down level sets overlaid upon the palmprint image used in sections 7.4.2 and 7.4.3.

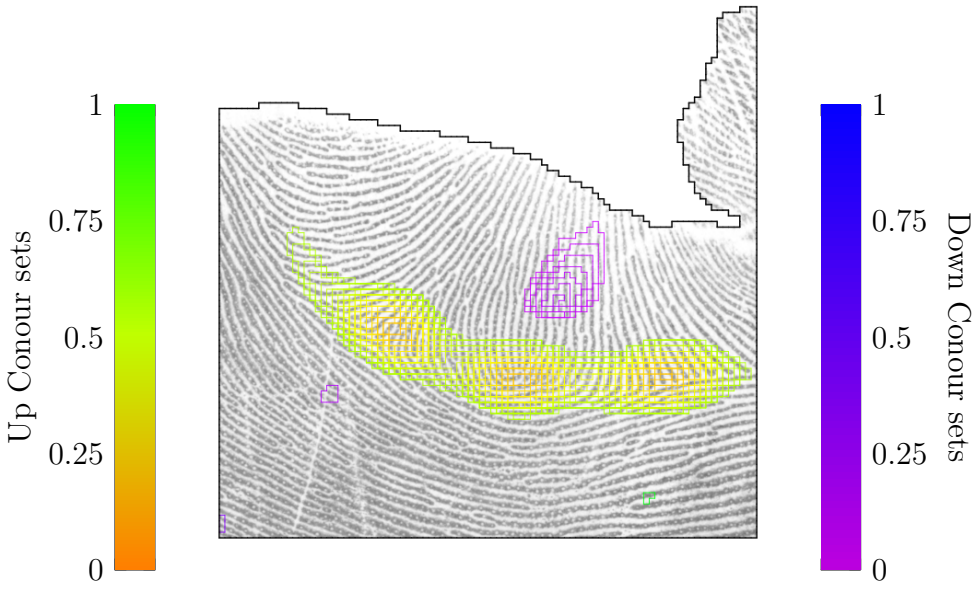


Figure 7.23.: Up and Down level sets

Figure 7.23 illustrates both the up and down level sets overlaid upon the full palmprint image used in the introduction to this section.

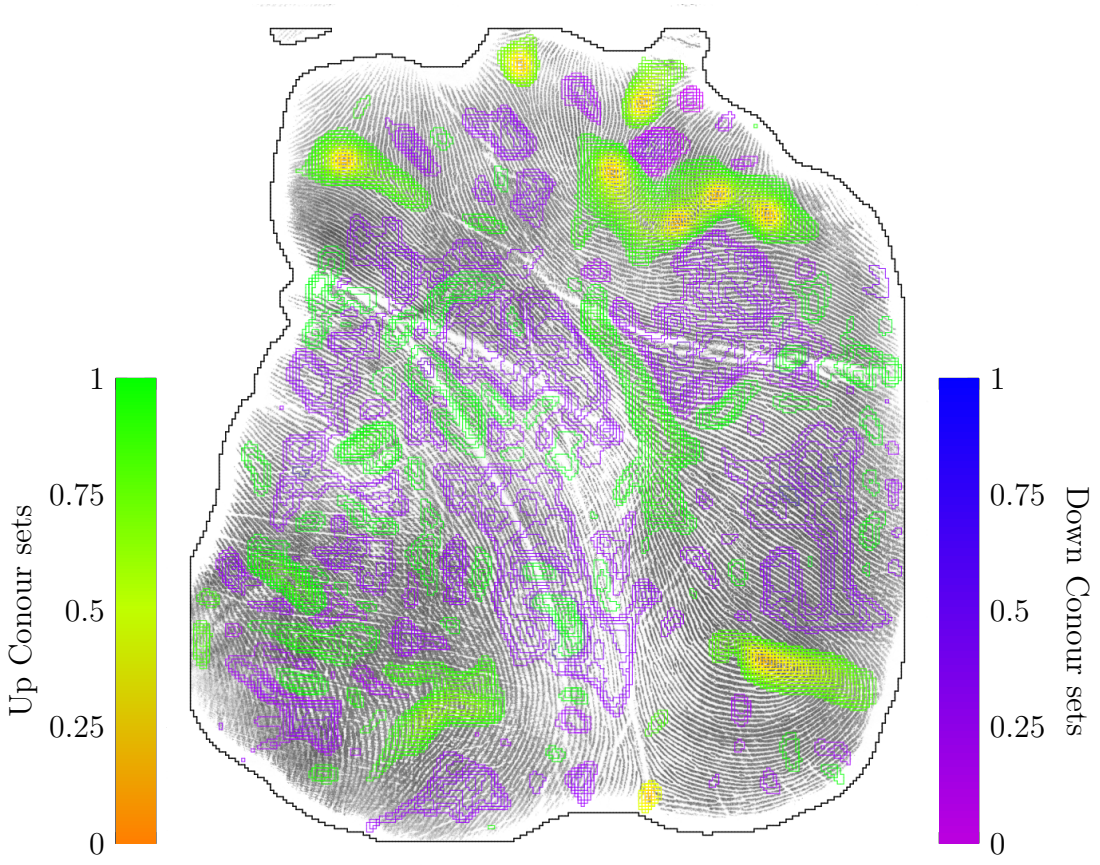


Figure 7.24.: Whole palm, Up and Down level sets

Both figures 7.23 and 7.24 illustrate that the up and the down level sets are hetero-

geneous (i.e not quite complementary). The heterogeneous properties to the up and down sets are useful for matching purposes and hence, a larger proportion of the palm is segmented.

7.4.5. Metrics for Each Region

Each regon within the palm is characterised using metrics which are used in the matching process. The metrics are categorised below in terms of geometry of the region, its circular statistics and structural relationship between the regions.

1. Physical size and shape of the independent regions:
   a) Area
   b) Centroid position
   c) Perimeter
   d) The seven Hu moments [93]
2. Circular statistics for each particular region:
   a) Mean orientation
   b) Variance
   c) Skew
   d) Kurtosis
3. Structural information of the tree:
   a) The level of the region
   b) The label of any super region
   c) The type of node (leaf, branch)
   d) The number of nodes contained within each region
   e) The number of leaves contained within the sub tree

Each of the metrics provides a unique value to a region, which is also invariant to rotation with respect to its dominant axes.

To calculate the metrics of each region, a binary mask $R(i, j)$ which represents the position of each of the pixels within the region is created:

$$R(i, j) = \{U(i, j, n) = r\} \tag{7.13}$$

Where n is the level at which the region exists and r is the regions numerical label.

Geometric Metrics

(a) The area of the region is calculated using:

$$\text{Area} = \sum R(i, j)$$

(b) The centroid position is calculated using:

$$\text{Centroid position} = \frac{(\sum i_n, \sum j_n)}{Area}$$

Where i_n and j_n are the i and j coordinates of each valid pixel from $R(i, j)$.

(c) The perimeter of a region, relies upon the definition of four-connected, referred as 4c below:

$$\text{Perimeter} = \sum \{(4c(R(i, j)) < 4) \cap (4c(R(i, j)) > 0)\}$$

(d) The final size and shape statistics are the seven Hu moments. The seven Hu moments are calculated using $R(i, j)$ and the methods in [93].

Circular Metrics

The four circular statistics are calculated from the set $\theta_r(i, j)$ which is the product of $R(i, j)$ and $\theta_{fr}(i, j)$:

$$\theta_r(i, j) = R(i, j)\,\theta_{fr}(i, j)$$

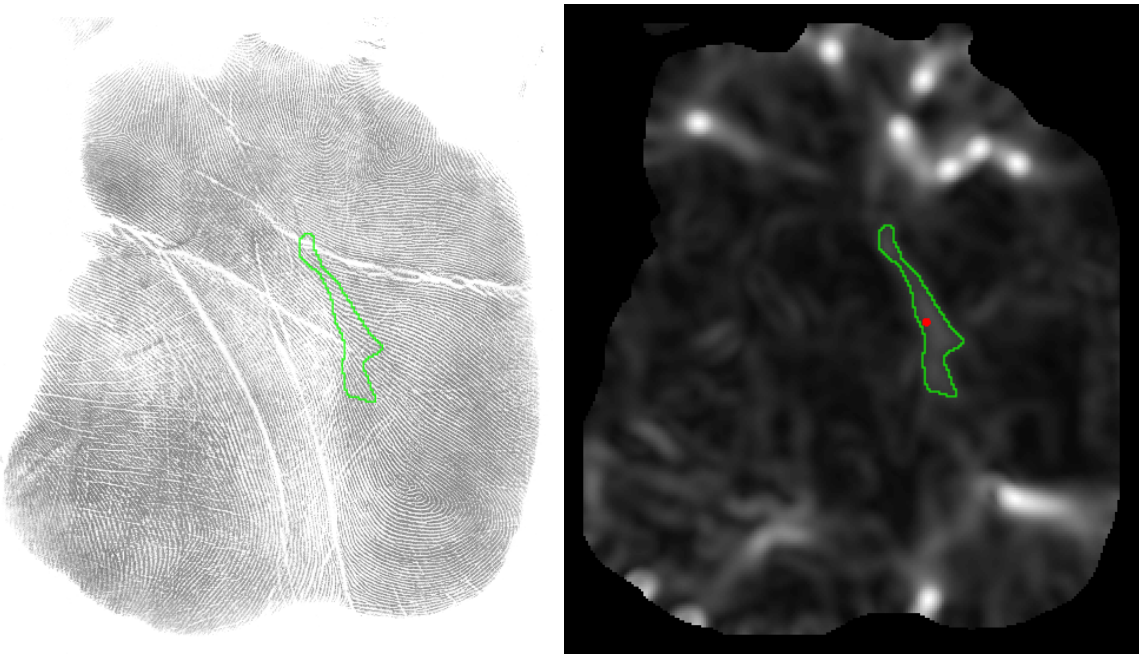
So $\theta_r(i, j)$ is the friction ridge orientation field estimates $\theta_{fr}(i, j)$ contained within the region $R(i, j)$. The mean orientation, variance, skew and kurtosis are calculated using equations 4.19, 4.12, 4.16 and 4.17 on the axial data $\theta_r(i, j)$.

Structural Metrics

The structural metrics of the tree are calculated as follows. The level of the region is known (i.e. n), the label of the parent node for every node was stored from section 7.4.2.1, these metrics are stored for every region. Using the tree structure, it is possible to scan through the regions to find the type of node, number of sub levels and the number of leaves contained within each sub tree of each region.

7.4.5.1. Example of the Region Metrics

To illustrate the metrics above, consider the region illustrated in figure 7.25, which is the region labelled 582, which exists in $U(i, j, n = 20)$.



(a) Region overlaid upon the palm (b) Region overlaid upon the circular variance

Figure 7.25.: The 582^{nd} region from the up level sets in $U\,(i,\,j,\,n=20)$

Therefore, a mask of this region (denoted, M_{582} and illustrated in figure 7.26 (b)) can be calculated by:

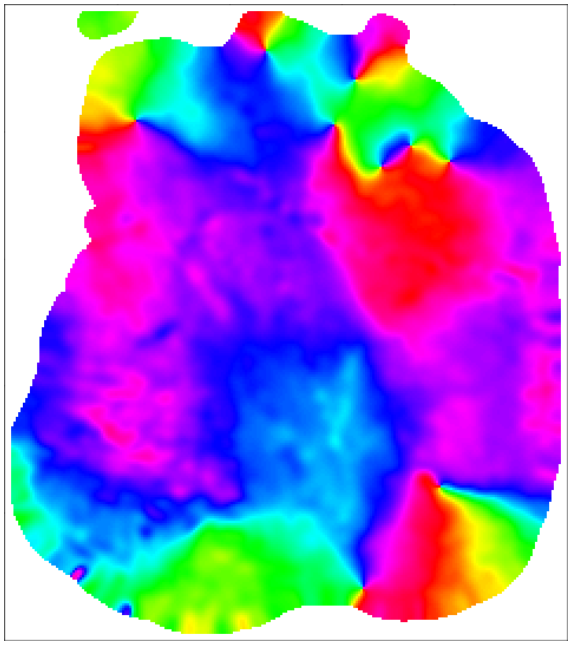
$$M_{582} = \{U\,(i,\,j,\,n=20) = 582\}$$

By using M_{582}, the geometric metrics can be calculated. The regions area is 626 units, the centroid position is (132.63, 151.14) to 5 significant figures and is illustrated by the red dot in figure 7.25. The seven Hu moments are 0.6646, 0.3744, 0.1094, 0.06937, 0.006018, 0.0387 and 0.0005557 to 4 significant figures, respectively. The perimeter of the region is 153 units.

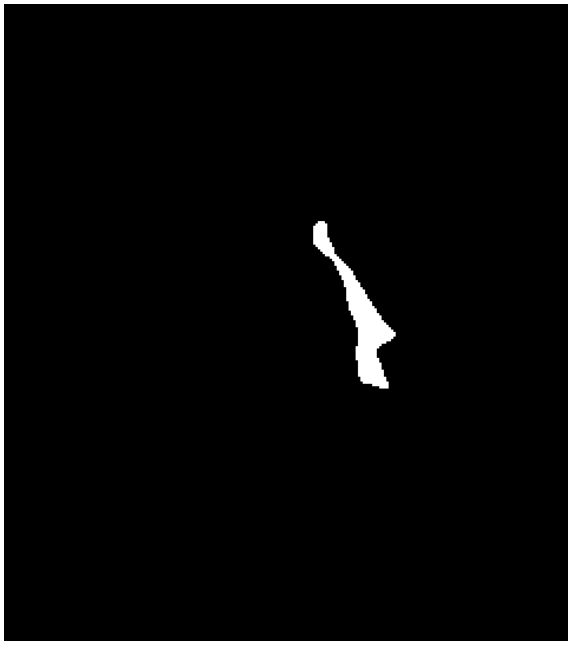
Figure 7.26 (a) illustrates the orientation field of the friction ridges $\hat{\theta}\,(i,\,j)$. The regions orientation field is calculated using equation 7.14,

$$M_{582,\,\theta} = M_{582} \cdot \hat{\theta}\,(i,\,j) \tag{7.14}$$

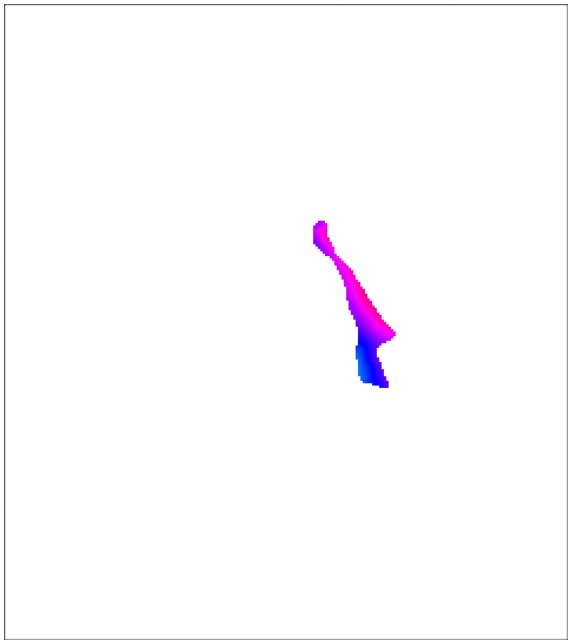
$M_{582,\,\theta}$ is illustrated in figure 7.26 (c), from which the circular statistics of the region calculated which are illustrated in figure 7.26 (d). The circular statistics of region $M_{582,\,\theta}$ are as follows: $\hat{s} = 0.05952$, $\hat{k} = 0.5679$, $\bar{R} = 0.9677$ and $\bar{\theta} = 2.4298$.



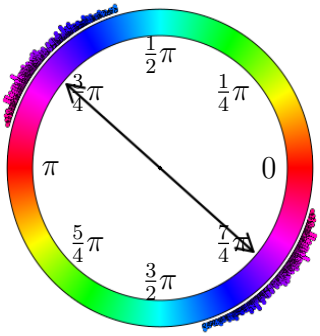
(a) $\hat{\theta}(i, j)$



(b) M_{582}



(c) $\theta_r(i, j)$



(d)

Figure 7.26.: Calculating $\theta_r(i, j)$

In the following chapter, two matching techniques are described, one for level I structures and one for the homogeneous level sets.

Chapter 8

Matching

8.1. Introduction

Since one of the aims of this thesis is to match latent palmprints of an unknown orientation, matching techniques which are invariant to both position and rotation must be developed. This chapter describes matching techniques which are invariant to translation and rotation for the features derived in chapter 7. The chapter consists of two main sections:

1. **Matching Level I structures** (section 8.2)
   In which individual matching procedures are developed for loops and deltas, based upon their *paths* represent as a list of Cartesian coordinates. The section creates invariant features for the lists, which are then used to match the individual structures.

2. **Matching structural homogeneous regions** (section 8.3)
   In which the regions identified in section 7.4 and structured in the *up and down trees,* are matched against the latent to form a ranked list. This list is then used to build a canvas representing the best match, in terms of its trees. The final matching score is determined by the correspondence between the canvas and latent.

The results of the individual matching methods are used separately in chapter 9, to derive the final ranked list of matches, which would be used in the comparison stage of the *ACE-V* procedure.

Consider a structure from a latent palmprint, known as a query. This query structure is matched against every structure of the same kind in the exemplar database, resulting in a matching score for every structure in the exemplar database. This allows the results to be ranked by the matching score, which results in a list of exemplars which best match the query.

8.2. Matching level I Structures

The matching techniques described in this section are based upon the characterisation of the friction ridge paths of loops and deltas, using a *path matching* technique. Since deltas and loops have a different numbers of paths, separate methods are required for each type of structure. The method is denoted Level I Structure Matching (LISM). Before these method can be introduced, *path matching* is explained.

8.2.1. Path Matching

In section 7.2.5, a path tracing method was introduced, which represented the path as a list of *absolute* image coordinates. As previously stated, lists of Cartesian coordinates are neither invariant to position or rotation. The following section describes how the list is transformed in to circular data to produce a set of discriminant invariant features, to be used in the structure matching procedure.

8.2.1.1. Invariant Features

To motivate the transformation from coordinates to invariant features, let us examine the three paths of the delta in figure 8.1.

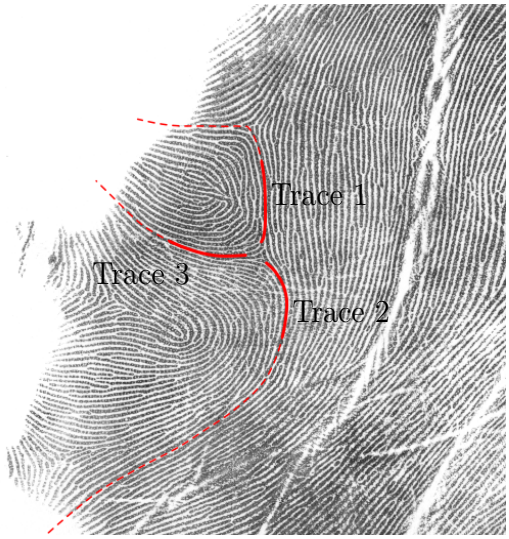


Figure 8.1.: Three paths of a delta (red)

The paths of the delta in figure 8.1 are illustrated by the three red lines. At each point along a path it is possible to calculate the relative angle subtended between two adjacent points (as shown in figure 8.2).

If this is done for all points along the path, the resultant list of circular data is invariant to translation. The formal method is described below.

Consider a path, $T_n = (x_n, y_n)$ for $n = 1, 2, .., m$ and where m is number of points in the path, represented in Cartesian coordinates, as shown in figure 8.1. The angle, $\theta(T_n)$, subtended between the points n and $n+1$ (illustrated in figure 8.2) can be calculated as in equation 8.1:

$$\theta(T_n) = \text{atan2}\left((y_{n+1} - y_n),\ (x_{n+1} - x_n)\right) \in [-\pi,\ \pi) \tag{8.1}$$

Note that this is circular data, not axial data, since $\theta(T_n) \in [-\pi, \pi)$ represents a vector which has both direction and angle. Given that $\theta(T_n)$ for $n = 1, 2, .., m-1$, is a set of circular data, it is invariant to translation and can be thought of as a continuous form of chain code [94].

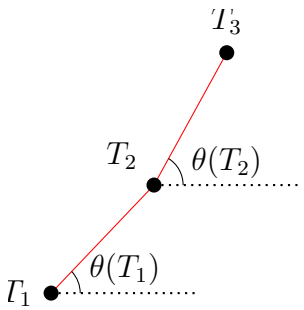


Figure 8.2.: The angle, $\theta(T_n)$, subtended between the points T_n and T_{n+1}

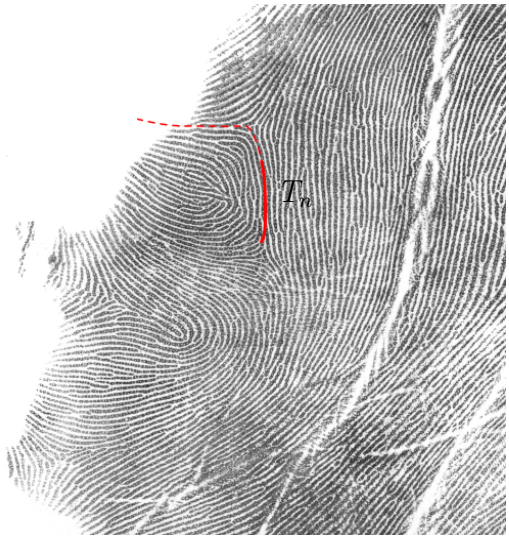
The path $\theta(T_n)$ can now be made invariant to rotation by subtracting the first angle in the path from *each* angle in the path. This is achieved by applying equation 5.1 to

$\theta(T_n)$:

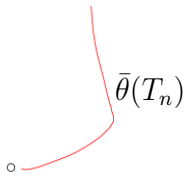
$$\overline{\theta}(T_n) = \text{atan2}\left((\theta(T_n) - \theta(T_1)), (\theta(T_n) - \theta(T_1))\right) \text{ for } n = 1, 2, .., m-1 \quad (8.2)$$

Where $\overline{\theta}(T_n) \in [-\pi, \pi)$. Effectively $\overline{\theta}(T_n)$ is now a path of length $m-1$ which begins with a direction of zero.

Figure 8.3 illustrates path T_n and its corresponding invariant path $\overline{\theta}(T_n)$. The start and end values of the Cartesian path and transformed circular data path are shown in table 8.1.



(a) A single path of a delta T_n



(b) The invariant path $\overline{\theta}(T_n)$, starting at (0, 0)

Figure 8.3.: A path from the delta and its invariant feature space

So far, this subsection has shown how the Cartesian coordinates of a path can be transformed into an set of circular data, which represents the angle made by moving a unit distance along the path.

n	Trace 1			
	x_n	y_n	$\theta(T_n)$	$\overline{\theta}(T_n)$
1	69.87	162.87	1.03	0
2	70.38	163.73	1.39	0.36
3	70.56	164.71	1.43	0.4
4	70.79	165.7	1.48	0.45
⋮				
55	40.70	191.41	2.92	1.89

Table 8.1.: The trace, orientation and path all to 2 d.p.

Both the delta and loop matching methods use the invariant paths for matching. Since there are a different number of paths which describe deltas and loops, the methods differ slightly.

The following two subsections describe the separate matching procedures for deltas and loops.

8.2.2. Delta Matching

Consider a delta ⅄ from an exemplar palm E (an example is illustrated in figure 8.4 (a)).

A delta is denoted as

$$E_{⅄_{i,n}} = (x_{i,n},\, y_{i,n}) \tag{8.3}$$

where,

$i = 1, 2, 3$ corresponding to each path of a delta

$n = 1, 2, .., m$ corresponds to the n^{th} point of a path, which is m points long

The delta is then processed, such that, for *each* path i, $\theta_{i,n}(E_{⅄})$ is transformed using equation 8.1,

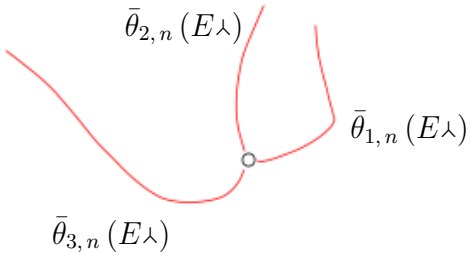
$$\theta_{i,n}(E_{⅄}) = \text{atan2}\left(\left(y_{i,n+1} - y_{i,n}\right),\, \left(x_{i,n+1} - x_{i,n}\right)\right) \;\in\; [-\pi,\, \pi) \tag{8.4}$$

so that the three traces of the delta, $\theta_{i,n}(E_{⅄})$ are invariant to translation. The path $\theta_{i,n}(E_{⅄})$ can now be made invariant to rotation using the method in section 8.2.1.1 as shown in equation 8.5.

$$\bar{\theta}_{i,n}(E_{⅄}) = \text{atan2}\left(\left(\theta_{i,n}(E_{⅄}) - \theta_{1,1}(E_{⅄})\right),\, \left(\theta_{i,n}(E_{⅄}) - \theta_{1,1}(E_{⅄})\right)\right) \tag{8.5}$$

Where $\bar{\theta}_{i,n}(E_{⅄}) \in [-\pi,\, \pi)$. But where $\bar{\theta}_{i,n}(E_{⅄})$ is now an array of three columns of length n where each column represents a path.

(a) Three paths of a delta $E_{⅄_{i,n}}$



(b) The invariant paths $\bar{\theta}_{i,n}(E_{⅄})$, where $\bar{\theta}_{1,n}(E_{⅄})$ starts from (0, 0)

Figure 8.4.: A path from the delta and its invariant feature space

Each of the invariant traces $\bar{\theta}_{1,n}(E_{⅄})$, $\bar{\theta}_{2,n}(E_{⅄})$ and $\bar{\theta}_{3,n}(E_{⅄})$ are illustrated by the red path lines in 8.4 and expressed in table form as shown in figure 8.2.

$\theta_{\mathbf{1},n}(E\curlywedge)$	$\theta_{\mathbf{2},n}(E\curlywedge)$	$\theta_{\mathbf{3},n}(E\curlywedge)$
0	⋮	⋮
⋮	⋮	⋮
⋮	⋮	⋮

Table 8.2.: Representation of an invariant delta, where each column corresponds a path

8.2.2.1. Matching a latent

Now, consider a second delta $L\curlywedge_{i,n}$ of unknown orientation (shown in figure 8.5 (a)) which is to be matched to $\bar{\theta}_{i,n}(E\curlywedge)$. $L\curlywedge_{i,n}$ is first transformed using equations 8.5 to give the , $\theta_{i,n}(L\curlywedge)$, which *only invariant to translation.*

Since the orientation of the delta $\theta_{i,n}(L\curlywedge)$ is unknown, there are three possible orientations in which $\theta_{i,n}(L\curlywedge)$ could match $\bar{\theta}_{i,n}(E\curlywedge)$. The three orientations are illustrated in figure 8.5 (b), (c) and (d).

To ease the notation for the matching metric, the notation for the three paths in $\theta_{i,n}(L\curlywedge)$ will be extended to $\bar{\theta}_{i,1,j}(L\curlywedge)$, where each of rotations in figure 8.5, is now represented by $j = 1, 2, 3$ and where $i = 1, 2, 3$ corresponds to the individual paths within each of the rotations. The representation of separates orientations are shown in table 8.3.

$j = 1$

$\theta_{i,n,\mathbf{1}}(L\curlywedge)$		
$\theta_{\mathbf{1},n}(L\curlywedge)$	$\theta_{\mathbf{2},n}(L\curlywedge)$	$\theta_{\mathbf{3},n}(L\curlywedge)$
$\theta_{\mathbf{1},\mathbf{1}}$	⋮	⋮
⋮	⋮	⋮
⋮	⋮	⋮

$j = 2$

$\theta_{i,n,\mathbf{2}}(L\curlywedge)$		
$\theta_{\mathbf{2},n}(L\curlywedge)$	$\theta_{\mathbf{3},n}(L\curlywedge)$	$\theta_{\mathbf{1},n}(L\curlywedge)$
$\theta_{\mathbf{2},\mathbf{1}}$	⋮	⋮
⋮	⋮	⋮
⋮	⋮	⋮

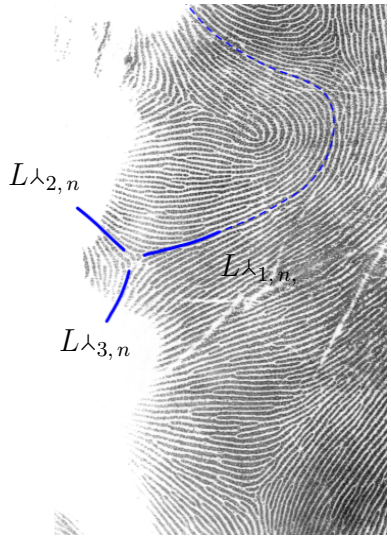
$j = 3$

$\theta_{i,n,\mathbf{3}}(L\curlywedge)$		
$\theta_{\mathbf{3},n}(L\curlywedge)$	$\theta_{\mathbf{1},n}(L\curlywedge)$	$\theta_{\mathbf{2},n}(L\curlywedge)$
$\theta_{\mathbf{3},\mathbf{1}}$	⋮	⋮
⋮	⋮	⋮
⋮	⋮	⋮

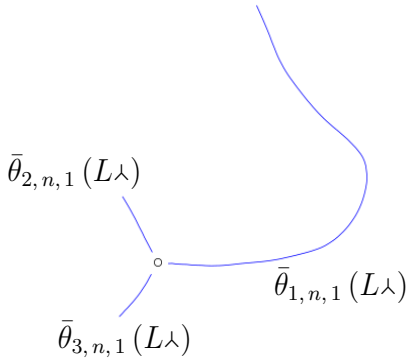
Table 8.3.: The three possible rotations of a latent delta

The deltas shown in table 8.3 are not invariant to rotation, it is now necessary to make the delta invariant to rotation. This is done by applying the rationally invariant operation in equation in 8.5 to each path within the table.

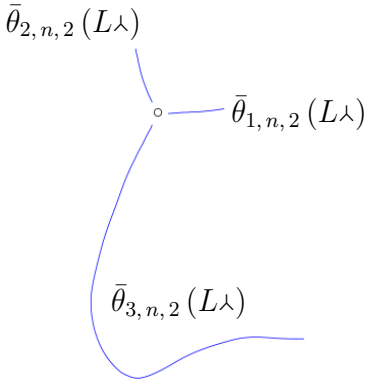
However, in this case the first angle of the first path (marked in blue in table 8.3) will be used as the reference in all applications of equation 8.5. Thus the delta represented in each table will be centred at the same point and a path within the delta will be aligned with a path within the exemplar delta. This is shown diagrammatically in figure 8.5.



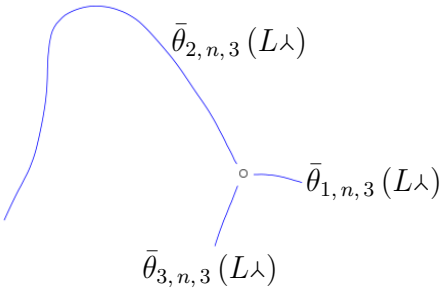
(a) Three paths of a delta $L\curlywedge_{i,n}$



(b) The invariant paths of $\bar{\theta}_{i,n,1}(L\curlywedge)$



(c) The invariant paths of $\bar{\theta}_{i,n,2}(L\curlywedge)$

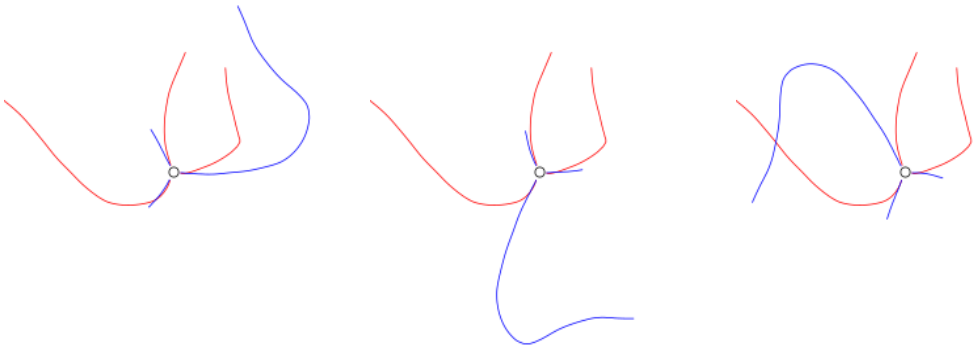


(d) The invariant paths of $\bar{\theta}_{i,n,3}(L\curlywedge)$

Figure 8.5.: The three possible sets of paths from the delta $L\curlywedge_{i,n}$ and their invariant feature space

These operations are defined is equation 8.6, where each sub-equation corresponds to the particular an operation on a specified table, $j = 1, 2, 3$.

$$
\begin{aligned}
j = 1; \quad & \bar{\theta}_{i,n,1}(L\curlywedge) = \operatorname{atan2}\left(\left(\theta_{\mathbf{1},\mathbf{1}}(L\curlywedge) - \theta_{i,n,\mathbf{1}}(L\curlywedge)\right), \left(\theta_{\mathbf{1},\mathbf{1}}(L\curlywedge) - \theta_{i,n,\mathbf{1}}(L\curlywedge)\right)\right) \\
j = 2; \quad & \bar{\theta}_{i,n,2}(L\curlywedge) = \operatorname{atan2}\left(\left(\theta_{\mathbf{2},\mathbf{1}}(L\curlywedge) - \theta_{i,n,\mathbf{2}}(L\curlywedge)\right), \left(\theta_{\mathbf{2},\mathbf{1}}(L\curlywedge) - \theta_{i,n,\mathbf{2}}(L\curlywedge)\right)\right) \\
j = 3; \quad & \bar{\theta}_{i,n,3}(L\curlywedge) = \operatorname{atan2}\left(\left(\theta_{\mathbf{3},\mathbf{1}}(L\curlywedge) - \theta_{i,n,\mathbf{3}}(L\curlywedge)\right), \left(\theta_{\mathbf{3},\mathbf{1}}(L\curlywedge) - \theta_{i,n,\mathbf{3}}(L\curlywedge)\right)\right)
\end{aligned}
\tag{8.6}
$$



(a) $j = 1$ (b) $j = 2$ (c) $j = 3$

Figure 8.6.: The three possible matches of $\bar{\theta}_{3,n}(E\curlywedge)$ and $\bar{\theta}_{3,n,j}(L\curlywedge)$

Now, to match an exemplar's delta $\bar{\theta}_{3,n}(E\curlywedge)$ to a latent's delta $\bar{\theta}_{3,n,j}(L\curlywedge)$, each of the paths from $\bar{\theta}_{3,n,j}(L\curlywedge)$ (i.e where $j = 1, 2, 3$) are independently matched to $\bar{\theta}_{3,n}(E\curlywedge)$. This process is illustrated in figure 8.6, where the exemplars delta path is red and the latents delta paths are blue.

Given the two sets of delta paths $\bar{\theta}_{3,n}(E\curlywedge)$ and $\bar{\theta}_{3,n,1}(L\curlywedge)$, the matching metric is calculated by taking angular difference (equation 4.1) between the respective paths of the exemplar and latent as shown in figure 8.6, where m is the cardinality of $\bar{\theta}_{3,n}(E\curlywedge)$ and $\bar{\theta}_{3,n,1}(L\curlywedge)$.

$$\curlywedge\left(\bar{\theta}_{3,n}(E\curlywedge), \bar{\theta}_{3,n,1}(L\curlywedge)\right) = \frac{1}{\pi m}\sum\left(\pi - \left|\pi - \left|\bar{\theta}_{3,n}(E\curlywedge) - \bar{\theta}_{3,n,1}(L\curlywedge)\right|\right|\right) \in [0,\, 1] \tag{8.7}$$

The metric is zero for an exact match. The matching process is performed three times (one for each rotation). The final score is calculated by taking the minimum of the individual scores (i.e the best match), as shown in equation 8.8.

$$\bar{\curlywedge}\left(\bar{\theta}_{3,n}(E\curlywedge), \bar{\theta}_{3,n,j}(L\curlywedge)\right) = \min\begin{pmatrix} \curlywedge\left(\bar{\theta}_{3,n}(E\curlywedge), \bar{\theta}_{3,n,1}(L\curlywedge)\right), \\ \curlywedge\left(\bar{\theta}_{3,n}(E\curlywedge), \bar{\theta}_{3,n,2}(L\curlywedge)\right), \\ \curlywedge\left(\bar{\theta}_{3,n}(E\curlywedge), \bar{\theta}_{3,n,3}(L\curlywedge)\right) \end{pmatrix} \tag{8.8}$$

This opertaion is performed for every delta in the exemplar database and ranked by increasing value, thus the closest match is the first in list.

8.2.3. Loop Matching

Since loops have only two paths a slightly different approach is required. The matching of loops can be simplified, since each individual loop has a primary angle. The primary angle, $E\Cap_\theta$, defined to be the angle subtended by the single friction ridge which flows directly out from the centre loop (section 7.2.5.4) and the zero direction.

Consider the loop in figure 8.7 (a), which has the two paths $E\Cap_{i,n}$ where $i = 1, 2$ specifies the path and n is the length of each particular path. The black point is the centre of the structure. The two paths which describe the loop are in red, each of which each starts at the blue dot and then move in opposite directions. The blue point

represents the point which is opposite the single friction ridge which flows directly out of the structure.

Like deltas, the result of the feature extraction (chapter 7.2) is a set of coordinates $E\Cap_{i,n}$ for each path (i.e. $i = 1, 2$). To make these paths invariant to translation each path is transformed, as before, using equation 8.1

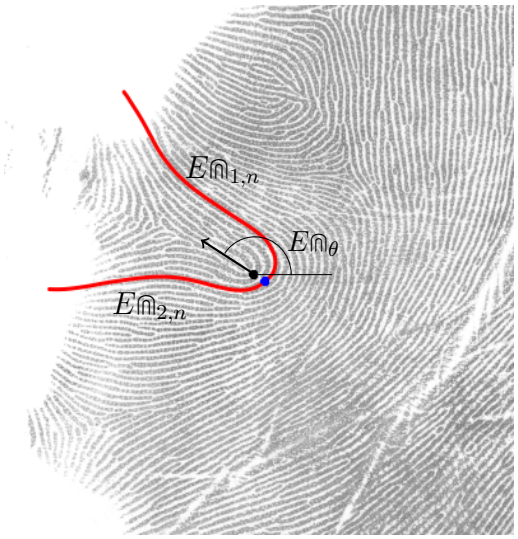
$$\theta_{i,n}\left(E\Cap\right) = \text{atan2}\left(\left(y_{i,n+1} - y_{i,n}\right), \left(x_{i,n+1} - x_{i,n}\right)\right) \bmod 2\pi \ \in \ [0,\ 2\pi) \tag{8.9}$$

Thus $\theta_{i,n}\left(E\Cap\right)$ for $n = \{1, 2\}$ are the two traces of the loop which are invariant to translation.

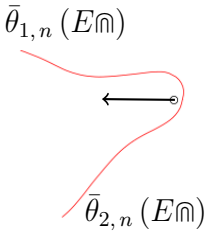
To ensure each of the paths is invariant to rotation, the angular distance ,$\bar{\theta}_{i,n}\left(E\Cap\right)$, between each angle and the normalised primary angle, $\pi - E\Cap_{\theta}$, is then calculated for each point on each path. Thus the new direction of the primary angle will always be π , as shown in figure 8.7 (b). This process is summarised in equation 8.10.

$$\bar{\theta}_{i,n}\left(E\Cap\right) = \text{atan2}\left(\left(\theta_{i,n}\left(E\Cap\right) - \left(\pi - E\Cap_{\theta}\right)\right), \left(\theta_{i,n}\left(E\Cap\right) - \left(\pi - E\Cap_{\theta}\right)\right)\right) \tag{8.10}$$

Thus $\bar{\theta}_{i,n}\left(E\Cap\right)$ can be represented as an array of *two* columns of length n, where each column contains each path and is now invariant to rotation.



(a) Two paths of a loop $E\Cap_{i,n}$ where the primary angle is 2.48 Radians.

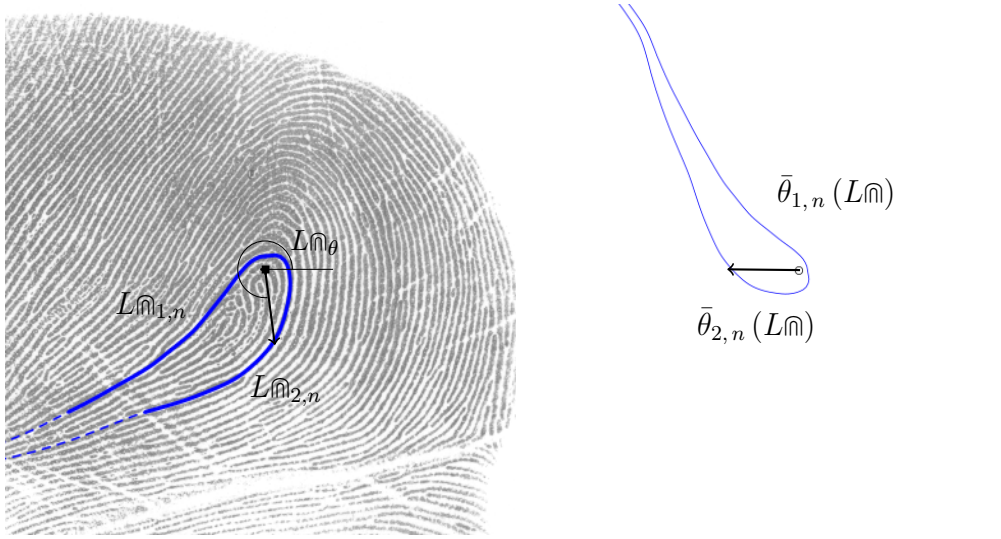


(b) The invariant paths $\bar{\theta}_{i,n}\left(E\Cap\right)$

Figure 8.7.: A loop and its invariant feature space

8.2.3.1. Matching Loops

Consider the case of a latent loop, $L\Cap$, which is to be matched against an exemplar loop, $\bar{\theta}_{i,n}\left(E\Cap\right)$, shown in figure 8.8. Figure 8.8 (a) shows a loop taken from the hypothenar, figure 8.8 (b) and the result of the application of equations 8.9 and 8.10, $\bar{\theta}_{i,n}\left(L\Cap\right)$.



(a) Two paths of a loop $L\Cap_{i,n}$ (b) The invariant paths $\bar{\theta}_{i,n}(L\Cap)$

Figure 8.8.: A loop and its invariant feature space

The matching metric is identical to equation 8.7, where however, in this case, it need only be applied once.

$$\Cap\left(\bar{\theta}_{i,n}(E\Cap), \bar{\theta}_{i,n}(L\Cap)\right) = \frac{1}{\pi m}\sum\left(\pi - \left|\pi - \left|\bar{\theta}_{i,n}(E\Cap), \bar{\theta}_{i,n}(L\Cap)\right|\right|\right) \in [0,\,1] \quad (8.11)$$

Where once again m is the cardinality of $\bar{\theta}_{i,n}(E\Cap)$ and $\bar{\theta}_{i,n}(L\Cap)$.

An example of matching is illustrated in 8.9, for dissimilar loop. The matching metric (equation 8.11) for $\bar{\theta}_{i,n}(E\Cap)$ (red) and $\bar{\theta}_{i,n}(L\Cap)$ (blue) is 0.4561 to 4 d.p.

So in the same manner as for deltas; if a loop exists in a latent palmprint, the loop $\bar{\theta}_{i,n}(L\Cap)$ is matched against every loop in the exemplar database using equation 8.11, and the result is stored. The resulting metrics are then sorted into a list of increasing value, then the closest matches are at the top of the list.

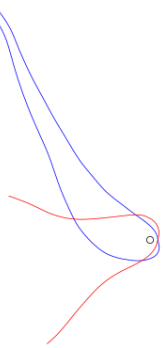


Figure 8.9.: Matching $\bar{\theta}_{i,n}(E\Cap)$ (red) and $\bar{\theta}_{i,n}(L\Cap)$ (blue), giving a matching score of 0.4561 to 4 significant figures.

8.2.4. Summary of Level I Structure Matching

Using the paths that describe structures (from section 7.2.5.3), this section has introduced an automatic matching procedure which is summarised by the following points:

1. A method has been developed that calculates a set of invariant features that describe the path of a structure: invariant to position and rotation.
2. These invariant features are used to match the paths of the structures.
3. Particular methods have been developed for different types of structure.
4. We match a structure to every structure in our database and rank the results.

The results of the matching methods applied to an exemplar database presented in chapter 9.

8.3. Cumulative Region Adding Algorithm

The previous section described how it was possible to match pairs of deltas and loops. However, a latent may not include either Level I structure and therefore, an additional matching technique is required to make use of the information contained in the remaining Level I detail (i.e. the information contained in Up and Down trees). The *Cumulative Region Adding Algorithm* (CRAA), performs matching of the Level I detail using region features and structure of the Up and Down trees.

8.3.1. An Overview of Cumulative Region Adding Algorithm

Given the regional features of a latent and a single exemplar palmprint, the latent features are matched against those of the exemplar to form a ranked list. The Cumulative Region Adding Algorithm then takes the ranked list and beginning with the candidate with the best score, iteratively adds latent regions to a latent canvas and exemplar regions to an exemplar canvas. Eventually, the exemplar canvas contains all the regions from the exemplar palmprint, which best match those of the latent. This canvas is then matched against the latent canvas. The overall process is shown in figure 8.11, which illustrates the iterative inclusion of *region pairs* to the respective canvasses. The example shown in figure 8.11 uses a relatively large latent to illustrate the implementation of CRAA, where the black line indicates the boundary of the valid palmprint area.

The Cumulative Region Adding Algorithm has the following steps:

1. **Rank possible individual region matches by score**. (Section 8.3.2)
   The matching procedure begins by isolating candidate regions within the exemplar, according to their variance. If the variance of a candidate is within an given interval, the candidate is processed further. The variance of an individual region corresponds to its height within the tree. Therefore this process is efficiently implemented by indexing into the corresponding tree.
   Further processing is then performed upon the candidate using a range of features. The features used differ according to whether an up tree or down tree matching is performed. The candidates are then ranked according to their combined up tree and down tree scores and classified as seeds, if the combined score is greater than a given threshold, to produce S_l and S_e. The characteristics of the seed region are used later in step 2, to guide which further matched regions are to be added to the canvas.

2. **For Each seed** α,

   a) **Seed Growth:** (Section 8.3.3)
   The latent canvas $C_{l,\alpha}$ and exemplar $C_{e,\alpha}$ are created of the same size as the latent and exemplar orientation fields. Both canvasses are initially blank (i.e. an array of NaN). **Add** the seed α, to both canvasses.

      i. **Select** the next highest ranked region pair, $S_{l,i}$ and $S_{e,i}$, from S_l and S_e and *temporarily* add the region pair to their respective canvas.

      ii. **If** *temporary* contents of the canvasses regions pass given criteria **Then** made the contents *permanent* additions **Else** *remove* $S_{l,i}$ and $S_{e,i}$ **End If**

      iii. **Repeat** (i), (ii) until the lists S_l,S_e are exhausted.

b) **Weight** each $C_{l,\alpha}$ according to by its area and number regions, to form its final score.

3. **End For**

4. **Select** the $C_{l,\alpha}$ with the largest score, as the best match that particular exemplar.

The next section describes the algorithm in detail.

8.3.2. Ranking Individual Region Matches

The ranking of paired candidate matches (region pairs) is performed by algorithm 8.1. In the algorithm RIRM, we denote the latent up and down region statistics as L_{ur} and L_{dr}, and the exemplar up and down region statistics as E_{ur} and E_{dr}, respectively.

The algorithm begins by creating two empty lists S_l and S_e to contain candidate latent and exemplar region pairs. These lists are filled with candidates in the following manner.

For each region in L_{ur}, if any region in E_{ur} exists within ± 6 levels of the latents region position within the tree, then the features of the two regions are matched using using equation 8.13, and placed in the lists S_l and S_e, together with their scores. The process is repeated for the down tree region (i.e. L_{dr} and E_{dr}) using equation 8.14.

The resulting lists of S_l and S_e of paired candidate matches are then ordered according to the individual scores. It should be noted that a region may match more than one other region and therefore, appear in S_l and S_e more than once.

Given this matching philosophy is greedy, the selection of first region pair and the order in which subsequent region pairs are added, is critical to the success of the algorithm.

The following section describes the regional feature matching and how this differs for up tree and down tree matching.

Algorithm 8.1 RIRM - Ranking Individual Region Matches

Require: $L_{ur,m}$, $L_{dr,n}$, $E_{ur,i}$ and $E_{dr,j}$ ▷ Sets of up and down region statistics for latent and exemplar, of length m, n, i and j, respectively.

Ensure: S_l and S_e ▷ Lists of matched region pairs

for i in 1:m **do**
  for j in 1:n **do**
    if $|\text{Level}(L_{ur,m}) - \text{Level}(E_{ur,n})| \leq 6$ **then**
      $M = urm\,(L_{ur,m},\, E_{ur,n})$
      $S_l = L_{ur,m},\, M$
      $S_e = E_{ur,n},\, M$
    end if
  end for
end for
for i in 1:p **do**
  for j in 1:q **do**
    if $|\text{Level}(L_{dr,i}) - \text{Level}(E_{dr,j})| \leq 6$ **then**
      $M = urm\,(L_{dr,i},\, E_{dr,j})$
      $S_l = L_{dr,i},\, M$
      $S_e = E_{dr,j},\, M$
    end if
  end for
end for
Order S_l and S_e by M

8.3.2.1. Individual Region Matching

This section describes the a metric used to match the two sets of region data. When matching using the up and down level sets, two subtly different algorithms are used.

The up tree is dominated by regions of high variance. The skew statistic (see chapter 4), which is used in down tree matching, is meaningless when the variance is high [72]. Generally, the down tree regions have low variance, therefore, the skew statistic is used exclusively in the down tree matching sets.

The score produced by the *up tree* matching equation 8.13, is an average of of four other sub-scores. The sub-scores are based upon the metric in equation 8.12, which is a measure of the similarity, where f corresponds to the feature in question and a and b are the scalar values of the feature.

$$\Psi_f(a,\, b) = \frac{\min(a,\, b)}{\max(a,\, b)} \tag{8.12}$$

The *up tree* metric is given by equation 8.13,

$$urm\,(L_{ur},\, E_{ur}) = \frac{1}{4}\left(\Psi_{area}(a,\, b) + \Psi_{perimeter}(a,\, b) + \Psi_{\bar{R}}(a,\, b) + \Psi_h(a,\, b)\right) \tag{8.13}$$

Where h is the height in of the level set within the tree.

The *down tree* metric is given by equation 8.14,

$$drm\left(L_{dr},\, E_{dr}\right) = \frac{1}{5}\left(\Psi_{area}(a,\, b) + \Psi_{perimeter}(a,\, b) + \Psi_{\bar{R}}(a,\, b) + \Psi_{h}(a,\, b) + \Psi_{\hat{s}}(a,\, b)\right) \tag{8.14}$$

Where both $urm\left(L_{ur},\, E_{ur}\right) \in [0,\, 1]$ and $drm\left(L_{dr},\, E_{dr}\right) \in [0,\, 1]$ and where 1 corresponds an exact match.

8.3.3. Seed Growth

The seed growth process is described in algorithm 8.2. Algorithm 8.2 processes the ordered lists of paired matches S_l and S_e and to produce a canvas for matching. The algorithm is potentially time consuming and so there are a nested sequence of criteria (lines 3-5), a failure of any of which will terminate the matching of that particular pair of regions.

The first of these criteria, rejects the candidate seed region, if its individual matching score is less than 0.95.

The second of the criteria, tests whether the difference between the areas of the region pair is less than 40% of the region whose area is largest.

The final criterion tests whether maximum absolute difference between the Hu moments of the region pair is less than 5%.

The second criteria effectively determines if the size and shape of the region pair is sufficiency similar to warrant further processing. If this is the case, this is performed in lines 6-7. If a region pair is accepted on to the canvasses all occurrences of that latent and exemplar regions are removed from S_l and S_e (shown at line 10 and 25). Pre-calculation and storage of the region's centroid takes place in line 8-9, for later use.

We will now describe the operation of the algorithm 8.2 in lines 11-25, in which it is determined whether the region pair should be permanently added to the canvasses. Lines 12-19 pre-calculate the values required for a test at line 20, which implements the final criteria for permanent inclusion on the canvasses. These final criteria are listed below and all criteria must be fulfilled for a region pair to be permanently placed on their respective canvasses.

Criteria for Inclusion of a Region Pair:

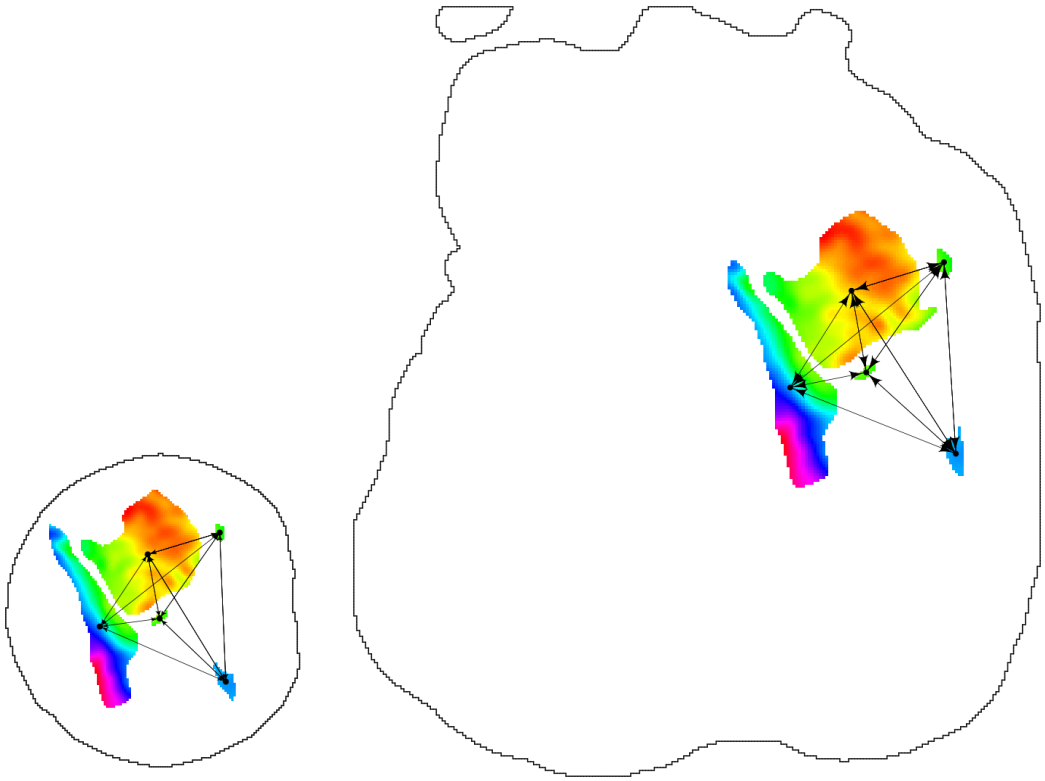
1. As a potential region pair is added to the canvasses, the distance between the centroid of the seed and the centroid of the new region is measured on both canvasses (see figure 8.10). For the region to be accepted the difference of the distances must be less that 40% of the largest of the two distances.

2. $\Psi_{area}\left(C_l,\, C_e\right) \geq 0.9$

3. $\Psi_{\bar{R}}\left(C_l,\, C_e\right) \geq 0.95$

4. $\Psi_{Hu}\left(C_l,\, C_e\right) \leq 0.05$

Where Ψ_{area}, $\Psi_{\bar{R}}$ and Ψ_{Hu} refer to the area, the circular variance from the resulting and Hu moments of the contents of the binary canvas, as the canvas of the final match is built up.



(a) **Latent**: distances between region centroids

(b) **Exemplar**: distances between region centroids

Figure 8.10.: An example of the distance between the centroids of matched region pairs. The final canvasses of CRAA, where the regions are shown as their orientation estimates to give the reader reference to visually identify region pairs. Each region has a dot in the centre which represents the centroid of each region. The length of each arrow represent the distances between each region. Notice the similar lengths of the corresponding arrows (from latent to exemplar), this is **Criteria 1** for the inclusion of a region pair, which must be fulfilled for canvas inclusion.

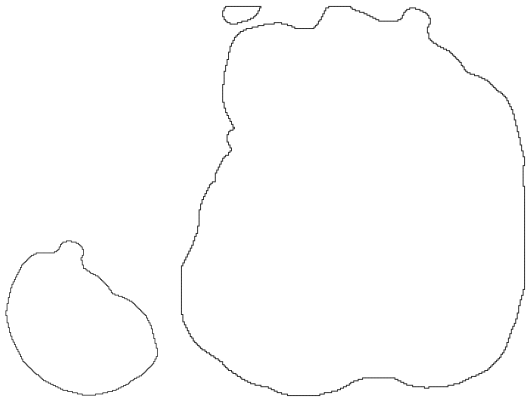
Algorithm 8.2 Seed Growth

Require: S_l, S_e ▷ Ordered lists of region pairs
Ensure: C_l, C_e ▷ C_l, C_e are the canvasses
Create $C_{l,\alpha}$, $C_{e,\alpha}$ ▷ $C_{l,\alpha}$, $C_{e,\alpha}$ are the empty canvasses for seed α
for i in $1:\alpha$ **do** ▷ α The number of Seeds
  if RIRM$(S_{l,i}, S_{e,i}) \geq 0.95$ **then**
    if $\Psi_{area}\,(C_{l,i},\, C_{e,i}) \geq 0.6$ **then**
      if Every $|\Psi_{Hu}\,(C_{l,i},\, C_{e,i})| \leq 0.05$ **then**
        $\mathrm{DL}_{x,y} = (S_{l,i,x}, S_{l,i,y})$ [1]
        $\mathrm{DE}_{x,y} = (S_{e,i,x}, S_{e,i,y})$ [2]
          ▷ DL is a list containing the coordinates of the centroid of $S_{l,i}$
          ▷ DE is a list containing the coordinates of the centroid of $S_{e,i}$
        Remove corresponding regions from $S_{l,i}$ and $S_{e,i}$ [3]
        for j in 1 :length$(S_{l,i})$ **do**
          DLat=$\sqrt{(S_{l,j,x} - \mathrm{DL}_x)^2 + (S_{l,j,y} - \mathrm{DL}_y)^2}$
          DExe=$\sqrt{(S_{e,j,x} - \mathrm{DE}_x)^2 + (S_{e,j,y} - \mathrm{EL}_y)^2}$
          D=min(Ψ(DLat,DExe)) ▷ D is the min Ψ from all centroids
          $C_{l,j} = C_l + S_{l,j}$ ▷ region $S_{l,j}$ is added to canvas $C_{l,j}$
          $C_{e,j} = C_e + S_{e,j}$ ▷ region $S_{e,j}$ is added to canvas $C_{e,j}$
          A=$\Psi_{area}(C_l, C_e)$
          V=$\Psi_{\bar{R}}(C_l, C_e)$
          Hu= max $|\Psi_{Hu}\,(C_{l,i},\, C_{e,i})|$
          if D$\geq$0.6 **and** A$\geq$0.9 **and** V$\geq$0.95 **and** Hu$\leq$ 0.05 **then**
            $C_l = C_{l,j}$ ▷ make $C_{l,j}$ permanent
            $C_e = C_{e,j}$ ▷ make $C_{e,j}$ permanent
            concatenate $(S_{l,j,x}, S_{l,j,y})$ to $\mathrm{DL}_{x,y}$
            concatenate $(S_{e,j,x}, S_{e,j,y})$ to $\mathrm{DE}_{x,y}$
            Remove corresponding regions from $S_{l,i}$ and $S_{e,i}$ [1]
          end if
        end for
      end if
    end if
  end if
end for ▷ **Store** Final score=$C_{l,\alpha}\times$ no. of regions

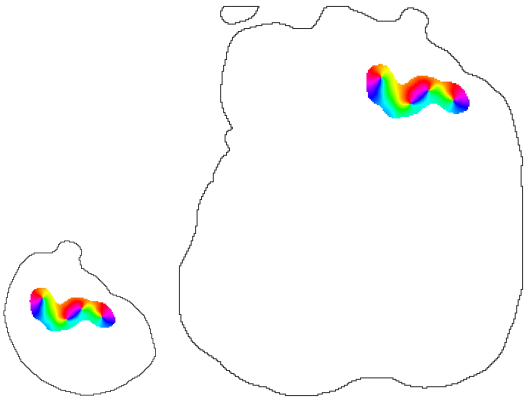
[1] Every entry of $S_{l,i}$ is removed from S_l, and any corresponding region pair is removed from S_e and every entry of $S_{e,i}$ is removed from S_e, and any corresponding region pair is removed from S_l
[2] $(S_{l,j,x}, S_{l,j,y})$ are the coordinates of the centroid of $S_{l,j}$
[3] $(S_{e,j,x}, S_{e,j,y})$ are the coordinates of the centroid of $S_{e,j}$

8.3.4. Example of Operation

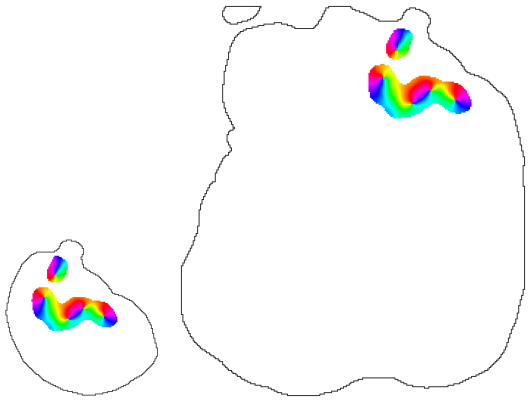
To illustrate the steps of the Cumulative Region Adding Algorithm, the added regions (shown by their orientation estimates) are overlaid upon the canvas's of a latent and an exemplar palmprints, which are illustrated in figure 8.11.



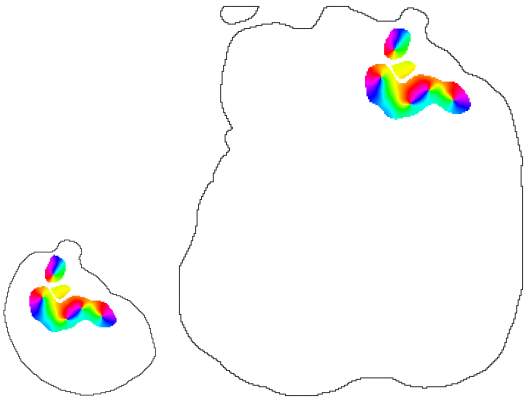
(a) **Step 2(a)0**, empty canvas of latent (left) and exemplar (right)



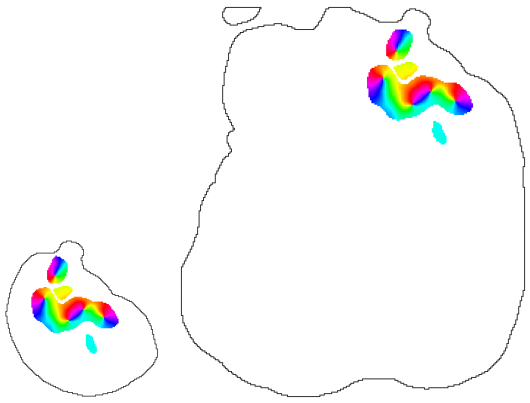
(b) **Step 2(a)**, first region pair (i.e. seed α) added to the blank canvasses



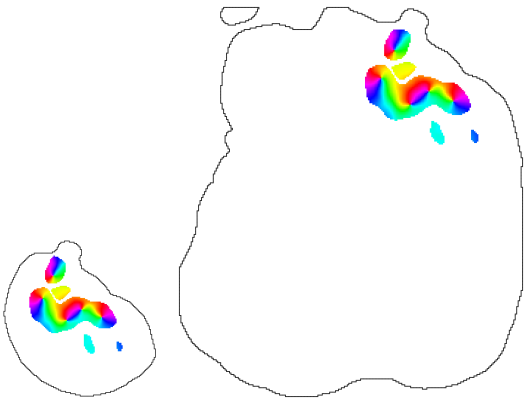
(c) **Step 2(a)ii**, the *first* region pair which passes the criteria, hence is added to the canvasses



(d) **Step 2(a)ii**, the *second* region pair which passes the criteria, hence is added to the canvasses



(e) **Step 2(a)ii**, the *third* region pair which passes the criteria, hence is added to the canvasses



(f) **Step 2(a)ii**, the final region pair which passes the criteria, hence is added to the canvasses

Figure 8.11.: CRAA, showing the iterative steps which add a region pair to the latent canvas (left of each sub-figure) and the exemplar canvas (right of each sub-figure). The black line indicates the boundary of the valid palmprint area for both cases of the latent and exemplar palmprints.

Chapter 9

Results

In ACE-V, the ranked matches from an APIS are manually examined during the comparison stage,. This chapter illustrates how the matching methods described in this thesis could aid a forensic examiner during the comparison stages of the ACE-V procedure. The exemplars in this section are taken from an anonymised inked palm data set. Since no *mated latent to full palmprint* databases are available in the public domain, the latents in this section have been created from the exemplar data set.

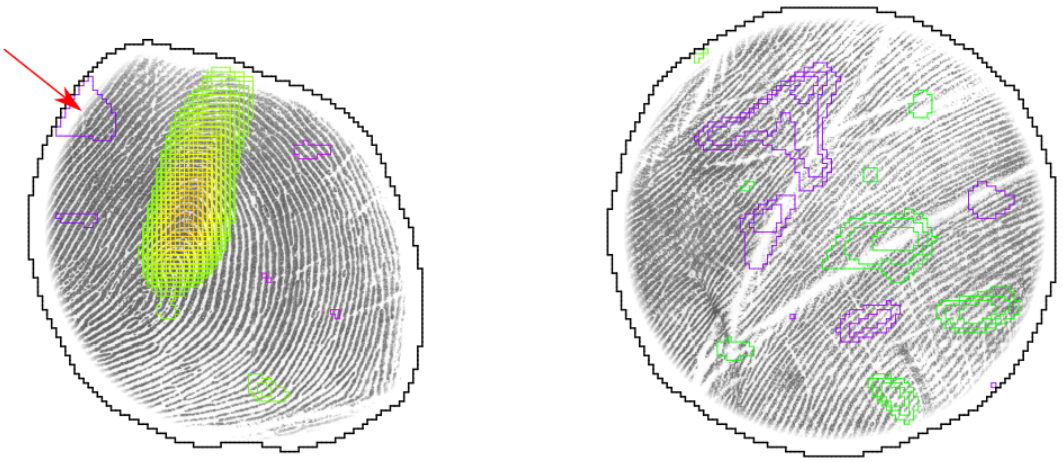
The structure of the chapter is as follows. Section 8.3 describes the results CRAA matching, when the pseudo-latents are derived from *any* part of the palm. Section 9.1 then refines the results to investigate the performance of CRAA in the *major regions* of the palm. Section 9.3, illustrates the matching performance of LISM when matching deltas and loops. Section 9.4 describes the effect of combining the results of CRAA and LISM when matching. In each of the sections, the effect of latent diameter is also investigated.

9.1. Matching Random Pseudo-Latents

The exemplar database, ED_i contains 226 palms. The pseudo-latents are created by randomly selecting a circular region from a random examplar. Examples of the feature extraction process applied to two pseudo-latents, are illustrated in figure 9.1.

Table 9.1 shows the results of attempting to match 1600 pseudo-latents of diameter of 33 mm using CRAA alone. The table shows the number of occurrences when CRAA correctly matched the pseudo-latent to the examplar and where this match occurred in the ranked list of matches. For example, the CRAA algorithm correctly matched 1251 latents, each of these matches had the highest matching score and so came came first in the ranked list of matches, 59 matches had the second largest matching score and so were second in the, list etc.

In ACE-V, the first n matches in the ranked list would be passed to the forensic investigator. Therefore, a useful measure of APIS performance on set of latents is, how often is a correctly matched latent, matched *with a rank less than* n. This would ensure that the sample size n, passed to the forensic examiner would include the correct match. Such a performance measure is the the cumulative rank of the results. The cumulative rank calculates how many of the correct matches from the latent set are contained within the first n of the ranked list. The rank and cumulative rank for the experiment above, is shown in table 9.1.



(a) 52 Up regions and 5 down regions (b) 14 Up regions and 10 down regions

Figure 9.1.: The features of random pseudo-latents. The boundaries of the Up regions are illustrated in green through red and boundaries of the Down regions are illustrated in purple through blue.

	Rank						
	1^{st}	2^{nd}	3^{rd}	4^{th}	5^{th}	10^{th}	...
No. of latents	1251	59	42	25	13	9	...
	78.2%	**3.7**%	**2.6**%	**1.6**%	**0.8**%	**0.4**%	...

	Cumulative Rank						
	1^{st}	≤ 2	≤ 3	≤ 4	≤ 5	≤ 10	...
No. of latents	1251	1310	1352	1377	1390	1439	...
	78.2%	**81.9**%	**84.5**%	**86.1**%	**86.9**%	**88.9**%	...

Table 9.1.: Frequency of ranked latent scores and the cumulative rank for 1600 pseudo-latents of diameter of $33mm$

The cumulative rank is illustrated, in blue, in figure 9.2, together along with a line which represents the total number of latents (shown by the blue dashed line). The error of the system for a particular n, can now be thought as the distance between the blue and dashed blue line. Therefore, we now see that CRAA ranked 78.2% of the correct matches in 1^{st} place, 86.9 of the correct matches were within the first five of the list (i.e ≤ 5) and the first 10 items in the ranked list contained 89.9 of all correct matches.

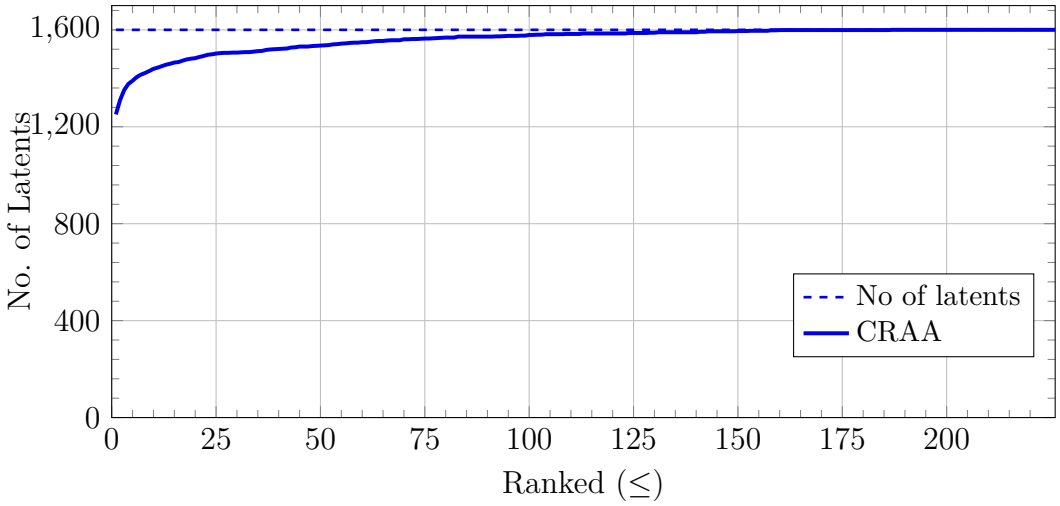


Figure 9.2.: Cumulative ranked score of CRAA

Removing Latents with low number of Homogeneous Regions

Figure 9.1 shows that number of homogeneous regions contained within a latent varies. Figure 9.3 is a histogram which illustrates the number of regions contained within each of the 1600 pseudo-latents.

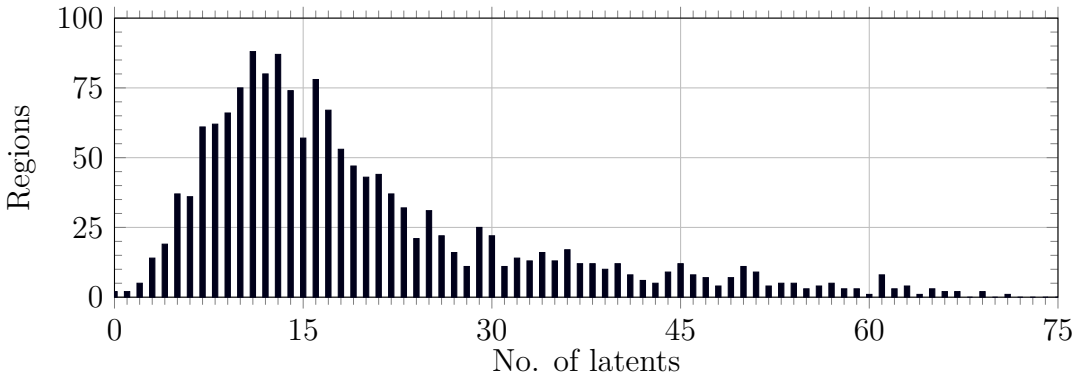


Figure 9.3.: The frequency of the regions contained in the 1600 latents

Figure 9.3 shows that a significant proportion (13%) of the pseudo-latents have ≤ 5 regions, reducing the chance of obtaining an accurate match [95].

By removing the pseudo-latents which containing fewer less than m homogeneous regions from the analysis, a more representative error rate can be calculated. Any latent which contains less than m regions is removed from the latent set and the previous experiment is repeated. Table 9.4 shows the cumulative rank for $m \geq \{1, 10, 20\}$, for the remaining 1219 latents., where it can be seen for latents with greater than 10 regions, 85.6% are ranked first. Alternatively, it can be seen for latents which contain more than 20 regions in this data set, there is a 90.3% chance of the correct match. For the sake of comparison, we will concentrate upon a nominal cumulative rank of $n \leq 10$, for the rest of the thesis.

	Cumulative Rank					
m	1^{st}	≤ 2	≤ 3	≤ 4	≤ 5	≤ 10
$\geq$1	78.2%	81.9%	84.5%	86.1%	86.9%	89.3%
$\geq$10	85.6%	88.8%	90.8%	92.2%	92.9%	94.9%
$\geq$20	**90.3%**	**92.6%**	**93.6%**	**95.2%**	**96%**	**97.4%**

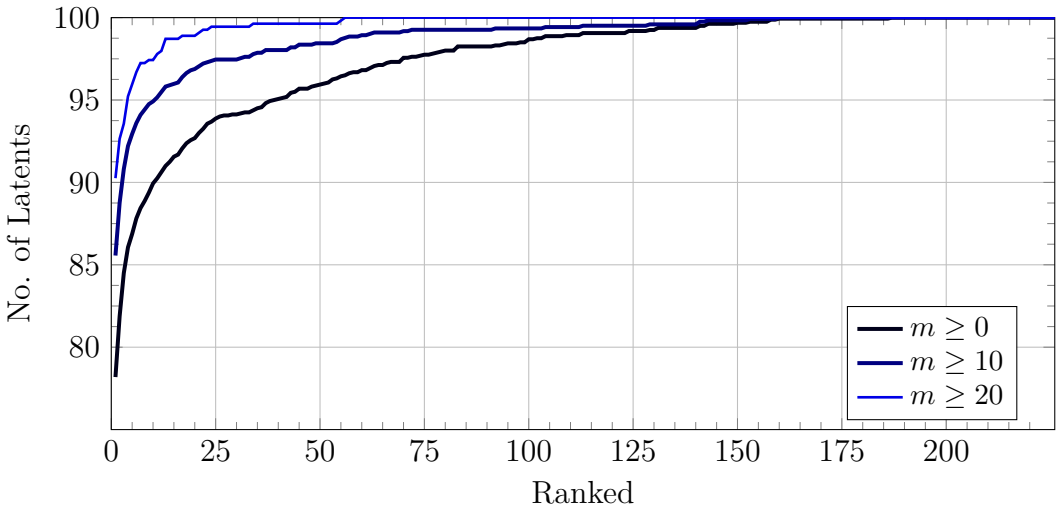


Figure 9.4.: Cumulative ranked score of **CRAA**, for the given minimum regions, m.

During the analysis of these results, it was observed that the accuracy of CRAA, was dependent upon which part of the palm (i.e. major region) the latent originated from. The following section repeats the previous analysis for latents, but for the cases where they derived solely from the triradiate hypothenar and thenar and for two diameters of latent.

9.2. Matching Random Pseudo-Latents from the Major Regions

Each palm in the exemplar database ED_i was manually split into its three *major regions* (i.e. *tri-radiate*, *hypothenar* and *thenar*) as illustrated in figure 9.5, to produce three *major regional databases*, each of which contains 226 images.

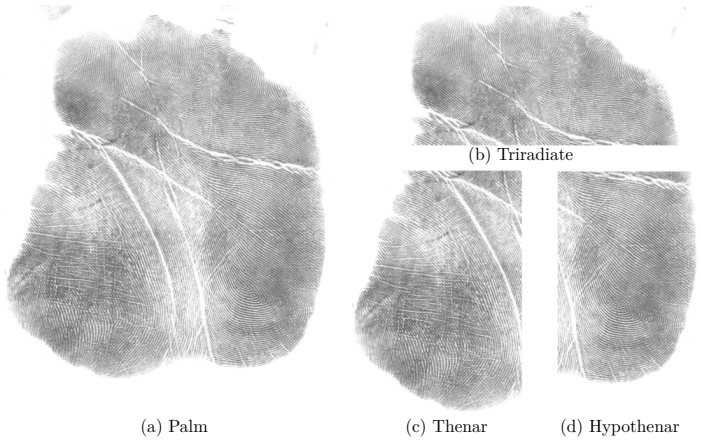


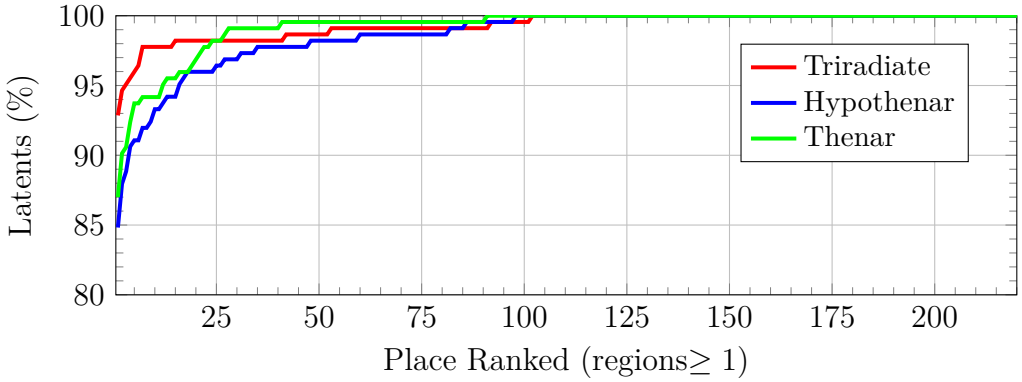
Figure 9.5.: Manually splitting a palm into its major regions

Random circular pseudo-latents of diameters $38mm$ and $25mm$ were then extracted from the three regional databases. Examples of major region pseudo-latents are shown at the top of figures 9.6 and 9.7. Each of the 226 pseudo-latents, from each of the the major region databases, were then matched against the exemplar database, using CRAA.

Figure 9.6 illustrates the cumulative rank of matches for each major region, for a pseudo-latent diameter of 38mm. Figure 9.7 illustrates the cumulative rank, when the latent diameter is 25mm.

From the table in figure 9.6, it is apparent that each of the corresponding cumulative ranks for $m \geq 1$ and $m \geq 10$, are within $\sim 1.5\%$, regardless of the region from which it was been derived and over 90% of the correct matches occur within the top 4. From the values marked in bold, it also apparent that latents from the tridate are the most discriminant.

Major Region	Cumulative Rank For regions $m \geq 1$, diameter $38mm$					
	1^{st}	≤ 2	≤ 3	≤ 4	≤ 5	≤ 10
Triradiate	**92.9%**	**94.6%**	**95.1%**	**95.5%**	**96%**	**97.8%**
Hypothenar	84.8%	87.9%	88.8%	90.6%	91%	93.3%
Thenar	87.1%	90.2%	90.6%	92.4%	93.8%	94.2%



Major Region	Cumulative Rank For regions $m \geq 10$, diameter $38mm$					
	1^{st}	≤ 2	≤ 3	≤ 4	≤ 5	≤ 10
Triradiate	**93.6%**	**95.4%**	**95.9%**	**96.4%**	**96.8%**	**98.6%**
Hypothenar	87%	89.4%	90.3%	92.3%	92.8%	95.2%
Thenar	87.3%	90.5%	91%	92.8%	94.1%	94.6%

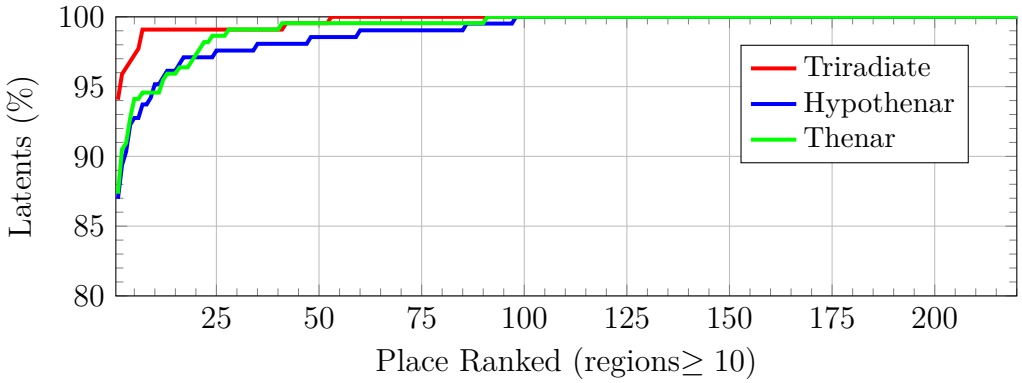
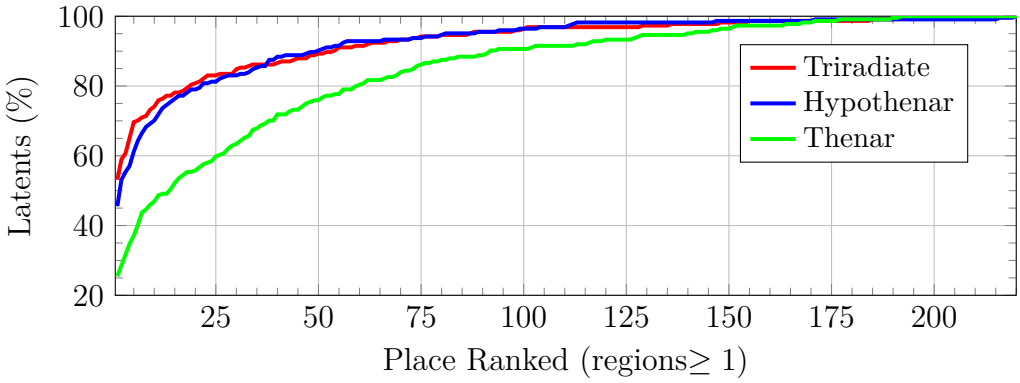


Figure 9.6.: The matching results of **CRAA** for pseudo-latents with a *diameter* of **38mm**, examples are on the top row from: *triradiate* (left), *hypothenar* (middle) and *thenar* (right). The top table and graph illustrate the cumulative rank for pseudo-latents with 1 or more regions, the bottom table and graph illustrate the cumulative rank for pseudo-latents with 10 or more regions.

Major Region	Cumulative Rank For regions $m \geq 1$, diameter $25mm$					
	1^{st}	≤ 2	≤ 3	≤ 4	≤ 5	≤ 10
Triradiate	**53.1%**	**58.9%**	**60.7%**	**65.2%**	**69.6%**	**74.1%**
Hypothenar	45.5%	53.1%	55.3%	57.1%	61.2%	70.1%
Thenar	22.3%	25.9%	26.8%	28.1%	29.9%	33.5%



Major Region	Cumulative Rank For regions $m \geq 10$, diameter $25mm$					
	1^{st}	≤ 2	≤ 3	≤ 4	≤ 5	≤ 10
Triradiate	**60.4%**	**67.4%**	**69.5%**	**74.3%**	**79.7%**	**82.9%**
Hypothenar	55.2%	62.8%	64.1%	66.2%	70.3%	79.3%
Thenar	27.6%	30.9%	34.3%	38.1%	40.3%	49.2%

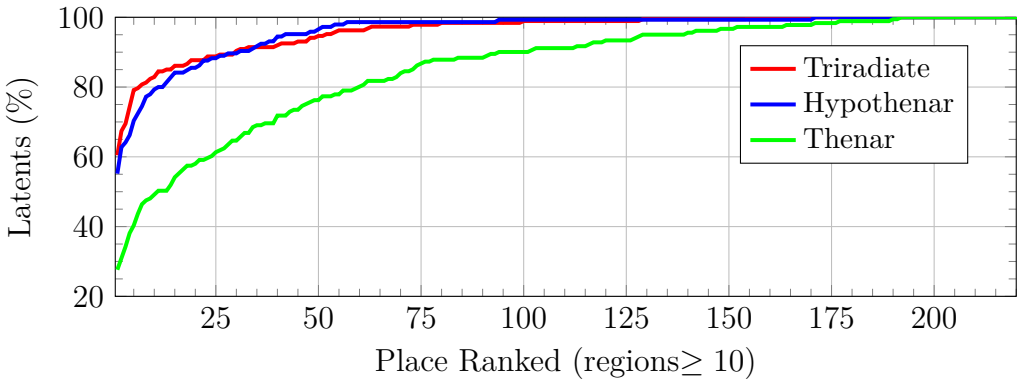


Figure 9.7.: The matching results of **CRAA** for pseudo-latents with a *diameter* of **25mm**, examples are on the top row from: *triradiate* (left), *hypothenar* (middle) and *thenar* (right). The top table and graph illustrate the cumulative rank for pseudo-latents with 1 or more regions, the bottom table and graph illustrate the cumulative rank for pseudo-latents with 10 or more regions.

For the smaller diameter latents, shown figure 9.7, it is apparent that each of the corresponding cumulative ranks for $m \geq 1$ and $m \geq 10$, are now significantly different. For

the case of $m \geq 10$, there is $\sim 10\%$ increase in values of the corresponding cumulative ranks. Once again the triradiate is the most discriminant.

However, when comparing the performance of thenar latents of diameter 38mm with that thenar latents of diameter of 25mm, we see a significant reduction in rank. The reduction varies, but can be as large a quarter, when matches are ranked first and approximately a half, when ranked within the top ten matches.

The results can be explained in terms of the typical circular variance of each major region's filtered orientation field. The variances of the major regions of the palm can be ranked as follows triradiate, hperthenar and thenar, where the thenar has the smallest circular variance (see section 3d). When interpreted in terms of region size, the homogeneous regions within the thenar are larger than those of the other major regions (see section 2.2.3 *long and short*). Therefore, when the latent diameter is reduced, not only are the number of bounded homogeneous regions reduced in thenar, so is their discrimination. The effect may also be compounded by the equal weight given to the number of regions and area, during Individual Region Matching section 8.3.2. This will be discussed further in chapter 10.1.

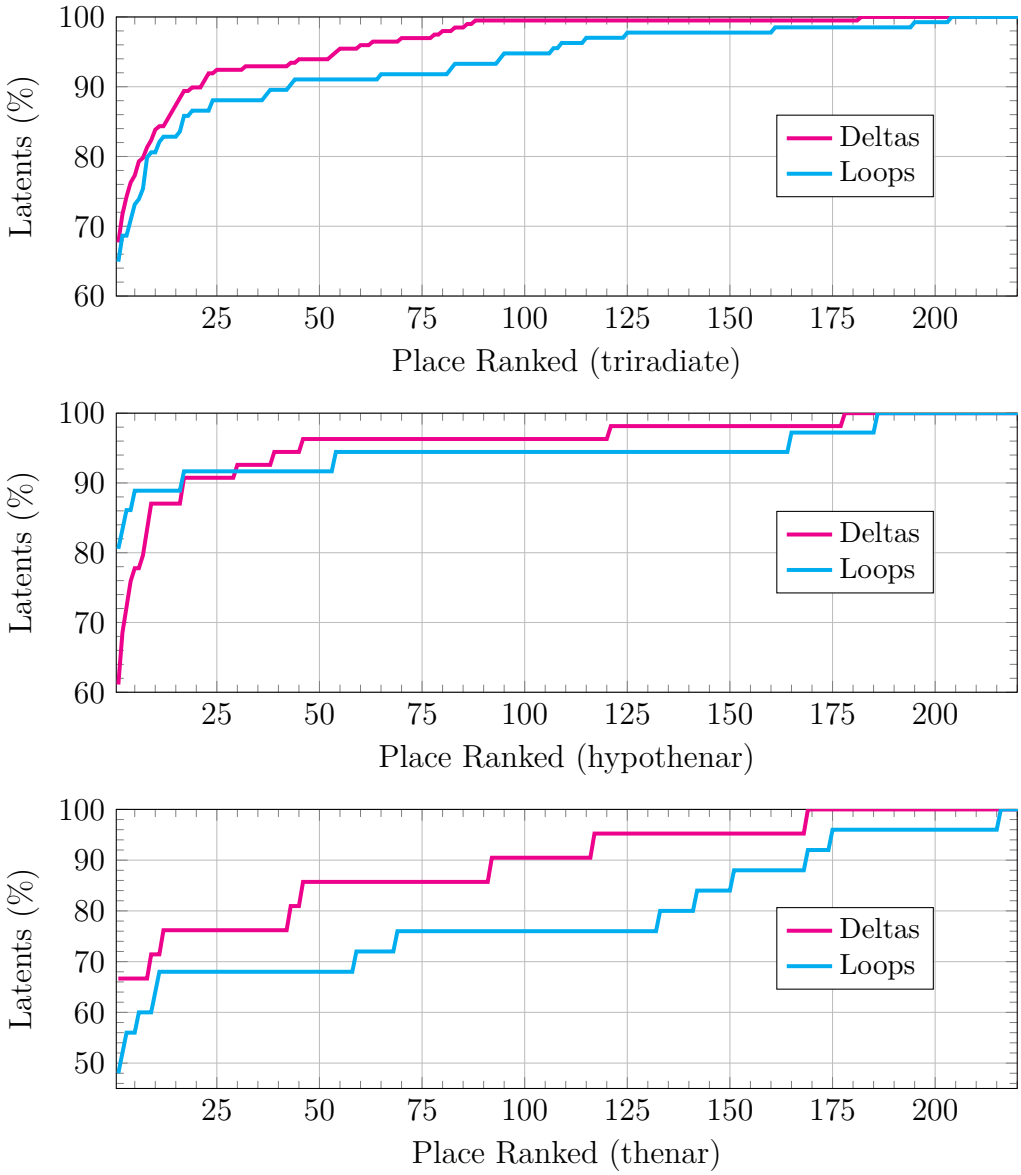
9.3. Matching Level I Structures

The previous analysis is now repeated using the Level I structure matching, as described in section 8.2. Figure 9.8 illustrates the results of Level I structure matching within the major regions of the palm, where the pseudo-latent is guaranteed to contain a loop or delta. Figure 9.8 shows the matching results for latents of diameter $38mm$. Figure 9.9 shows the results for latents of diameter of 25mm.

It is apparent from figure 9.8, that for latents of diameter 38mm, that loops and deltas within the hypothener are the most discriminate features within the palm. Although, the triradiate region contains the largest number such structures, there is a greater structural variation within the loops and deltas found in the thenar. This variation results in higher discrimination when used in the matching process. For example, a hypothenar may or may not contain a *higher carpel delta*, if the delta does exist, there will also be accompanying loops. This in contrast to the triradite, which always has at least three deltas and where often, the deltas share a similar structure.

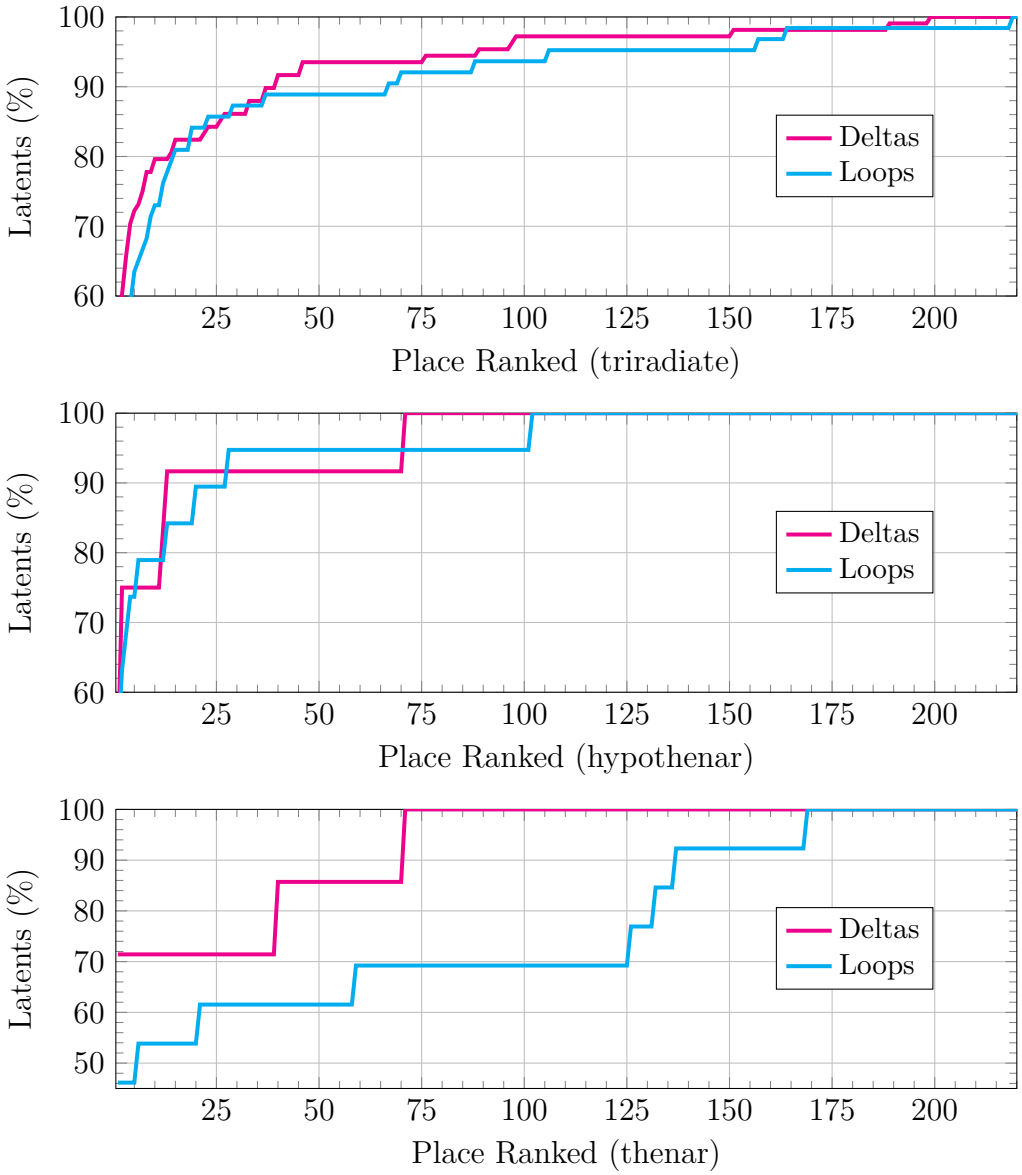
The thenar contains the fewest Level I structures, as indicted by table 7.2 in chapter 7. Figure 9.9 also indicates that it is not until 175 matches are included in the ranked list, that cumulative rank for the thenar approaches 90%. When taken in conjunction, these facts indicate that structures taken from the thenar have little discriminate ability, when compared to that of the triradiate and hypothenar.

When comparing the top ten cumulative rank for the smaller and the larger latents, the discriminate ability of the loops and deltas found in the hypothenar of the larger latents, is no longer apparent in the smaller latents. This further confirms the observation, concerning latent size and discriminant ability, made in section 9.2 for CRAA.



Major Region	Level I Structure	Cumulative Rank - diameter of 38mm					
		1^{st}	≤ 2	≤ 3	≤ 4	≤ 5	≤ 10
Triradiate	Delta	67.7%	71.7%	74.2%	76.3%	77.3%	83.8%
	Loop	64.9%	68.7%	68.7%	70.9%	73.1%	80.6%
Hypothenar	Delta	61.1%	68.5%	72.2%	75.9%	77.8%	87%
	Loop	**84.8%**	**87.9%**	**88.8%**	**90.6%**	**91%**	**93.3%**
Thenar	Delta	66.7%	66.7%	66.7%	66.7%	66.7%	71.4%
	Loop	48%	52%	56%	56%	56%	68%

Figure 9.8.: The matching results of Level I structure matching (**LISM**) for pseudo latents with a *diameter* of **38mm**, the graphs illustrate the results of : *triradiate* (top), *hypothenar* (middle) and *thenar* (bottom). The results are summarised in the table at the bottom.



Major Region	Level I Structure	Cumulative Rank - diameter of 25mm					
		1^{st}	≤ 2	≤ 3	≤ 4	≤ 5	≤ 10
Triradiate	Delta	55.6%	60.2%	65.7%	70.4%	72.2%	**79.6%**
	Loop	46%	49.2%	57.1%	58.7%	63.5%	73%
Hypothenar	Delta	50%	**75%**	**75%**	75%	75%	75%
	Loop	52.6%	63.2%	68.4%	**73.6%**	**73.6%**	78.9%
Thenar	Delta	**71.4%**	71.4%	71.4%	71.4%	71.4%	71.4%
	Loop	46.2%	46.2%	46.2%	46.2%	46.2%	53.8%

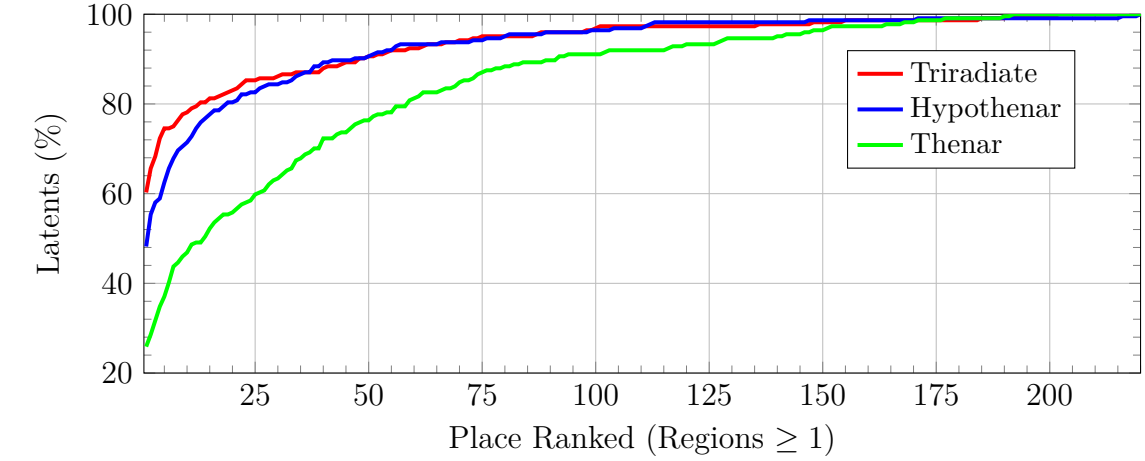
Figure 9.9.: The matching results of Level I structure matching (**LISM**) for pseudo latents with a *diameter* of **25mm**, the graphs illustrate the results of : *triradiate* (top), *hypothenar* (middle) and *thenar* (bottom). The results are summarised in the table at the bottom.

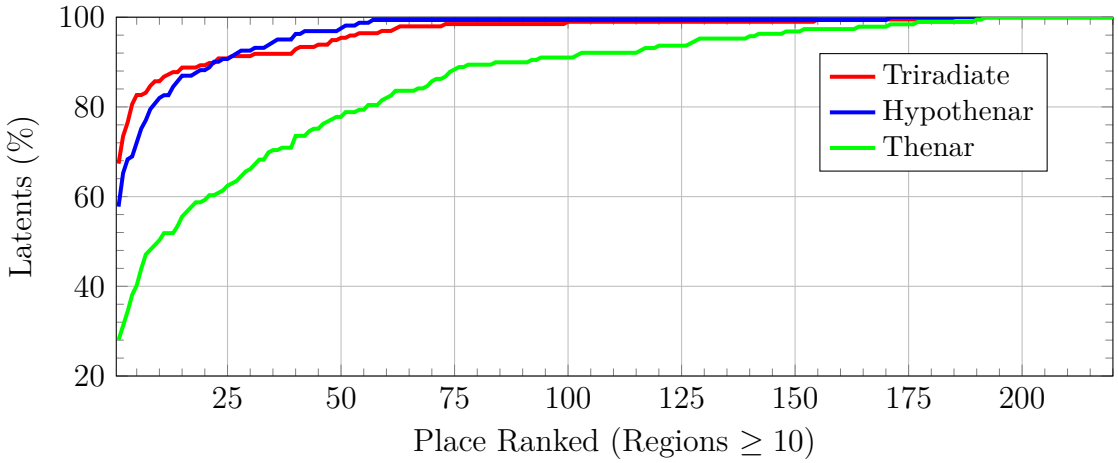
9.4. Combining CRAA and Level I Structure Matching

This section presents the results of a naive attempt to combine the rank of each matching technique. Both LISM and CRRA are applied to the exemplar database using the

same latents as in previous sections. The ranks from each technique are then combined by taking their minimum, this is then taken as the final rank of the match. ...

9.4.1. Matching Pseudo-Latents of size r=250

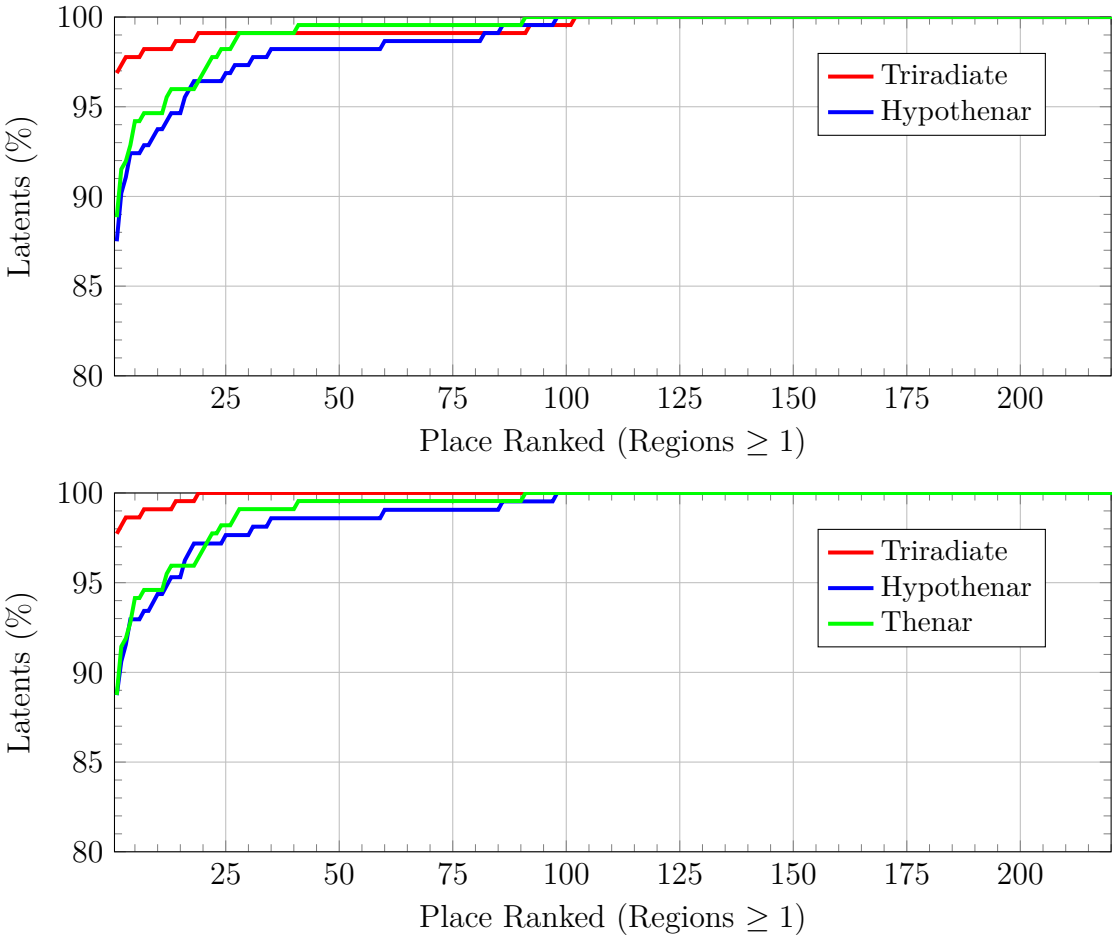




	CRAA		min(**CRAA**,**LISM**)	
Cumulative Rank	1^{st}	≤ 10	1^{st}	≤ 10
Triradiate	53.1%	60.4%	60.3%	67.3%
Hyperthenar	45.5%	55.2%	48.2%	57.8%
Thenar	22.3%	27.6%	25.9%	28%
All Regions	**40.3%**	**47.7%**	**44.8%**	**50.9%**

Figure 9.10.: Illustrates the combined affect of **CRAA** & **LISM** for a diameter of 25*mm* for each of the major regions. The table shows the percentage of matches, which ranked the match as first in the list of matches and those that were ranked within the first 10.

9.4.2. Matching Pseudo-Latents of size r=375



	CRAA		min(CRAA,LISM)	
Cumulative Rank	1^{st}	≤ 10	1^{st}	≤ 10
Triradiate	92.8%	97.8%	96.9%	98.2%
Hyperthenar	84.8%	93.3%	87.5%	93.8%
Thenar	87%	94.2%	88.8%	94.6%
All Regions	**88.2%**	**95.1%**	**91.1%**	**95.6%**

Figure 9.11.: Illustrates the combined affect of **CRAA** & **LISM** for a diameter of $38mm$ for each of the major regions. The table shows the percentage of matches, which ranked the match as first in the list of matches and those that were ranked within the first 10.

The effect of combining LISM and CRAA is shown in figure 9.10 and 9.11. It can seen that LISM has little effect upon whether the match is contained within the first 10 matches, however, it increases the likelihood that it will be the first ranked match (by approximately 3%). This result is expected, since wherever Level I structures exist, there will also exist many level sets (see section 7.4) and thus both CRAA and LISM would be expected to have higher scores in such areas.

However, what is unexpected, is that the results for the thenar latents of diameter $38mm$ are comparable with those of the other regions of the palm. The thenar contains large areas of cross hatching, which in previous techniques [27] has resulted in distortion of the orientation field, to such an extent that such techniques attempt to remove cross

hatching before generating the orientation field. The results from the thenar in table 9.11, illustrate the robust nature of the orientation estimate method described in in chapter 5 and 6.

Chapter 10

Conclusions and Further Work

The work contained within the thesis has developed techniques for the matching of *latent to full* palmprints using Level I detail. Hitherto palmprint matching techniques have been dominated by two approaches Firstly, *full to full* matching (e.g. [50, 17] and table 3.2 in chapter 3), which when applied to *latent to full* have significant reductions in matching accuracy. Secondly, *latent to full* in which the matching process is limited to the use of minutiae (e.g. [28, 39]). These have reduced matching accuracy in areas of higher circular variance (i.e about Level I structures and flexion creases) and in the thenar region. To the authors knowledge no previous study exists which has attempted *latent to full* matching using *purely Level I detail*.

The work has verified the feasibility of latent to full matching using Level I detail on a database of inked exemplars. The results show that it is possible to obtain matches within the ranked top ten, 95.6% of the time and ranked first, in 91.1% for latents with a $38mm$ diameter. The results go on to show that the cumulative rank of a match depends upon where the latent originated within the palm and the size of the latent. It is shown that regions of high circular variance such as tri-radiate are the most discriminant, where 96.9% are ranked first and 98.2% are ranked in the top ten. In addition, it is also shown that the matching of thenar latents is comparable to that of the tri-radiate and hypothenar, given an equivalent number of homogeneous regions exist within the latent. Thus the approach has been shown to compensate for two weakness in previous techniques, namely, the lack of matching accuracy in areas of high variance and within the troublesome thenar region.

The matching success decreases with a latent size $25mm$, to approximately 70% for the tri-radiate and hypothenar, and 30% for the thenar. By removing the latents with fewer than 10 homogeneous region there is a significant increase in accuracy for the tri-radiate and hypothenar. This provides a practical bound upon the current matching success as a function of the number of homogeneous regions contained within the latent. This is illustrated in figure 10.1. Therefore, the matching success of the system is determined by the number of homogeneous regions and not necessarily the size of the latent. The means by which the matching of fewer than 10 regions may be improved and the means by which regions broken by the boundary of the latent could be included in the individual region matching, are discussed below in section 10.1.

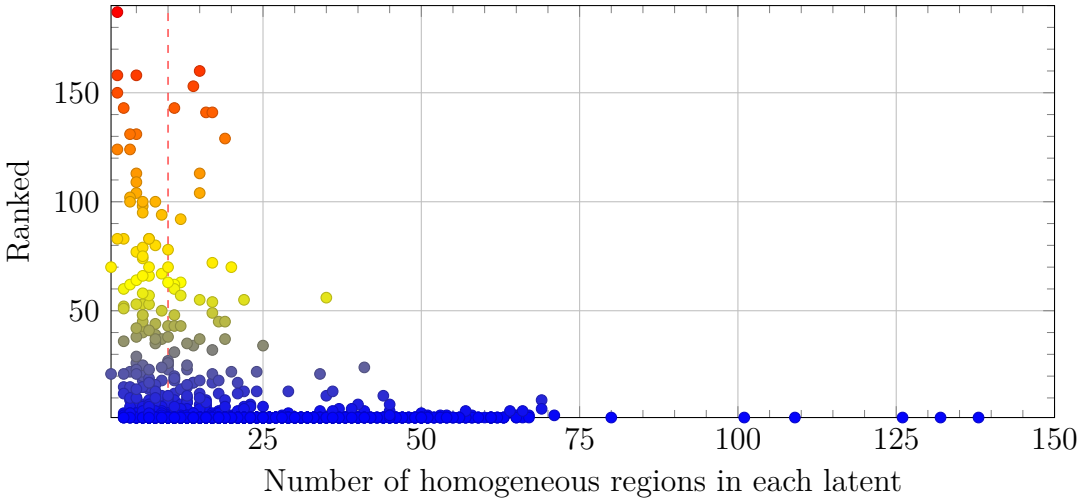


Figure 10.1.: The test results for the 1600 psuedo latents with a diameter of 33mm, where the rank of each psuedo latent is plotted against the number of homogeneous regions it contains. The red dotted line, represents 10 homogeneous regions. As the number of homogeneous regions increases, so does the latent's rank.

A method for the direct matching of Level I structures was also presented. The method created features, which are invariant to translation and rotation, based upon the paths which define the Level I structures. However, in this case, it was shown that the most discriminant latent were derived for the hypothenar, in which $93-80\%$ of the loops were within the top ten of the ranked matches. When naively combined with the CRAA matching technique, there was little change to whether the match was contained within the top ten. However, it did improve the rank of the correct match within the top ten.

During the development of the matching techniques a number of other results were found.

- That the use of circular statistics can be used to replace previous ad-hoc methods. Circular statistics, owing to their formal derivation, can be extended to higher order statics. It has been shown that higher order statistics, such as the variance, skew and kurtosis, can then to used to characterise Level I detail. This has three advantages. Firstly, they can be used to isolate *unreliable estimates* and filter the orientation estimate (Section 6.3). Secondly, the circular variance's maxima can be used to locate Level I structure (Section 7.2). Thirdly, they can used to identify and segment areas of homogeneous friction ridge flow (Section 7.4). Fourthly, they can be use during in matching, as a metrics to derive features which characterise isolated regions (Section 8.2).
- A method for the derivation of an optimal isomeric 3x3 kernel has been presented. The method specifies a specific ratio of the kernel for a specific wavelength. The performance of the optimal kernel was shown to exceed that of previous such kernels used in orientation estimation (Section 5.2).
- An improved orientation operator based upon spherical quadrature filters was developed in Section 5.3. A comparative study of SQFs was made and the *LoG* filter was shown to have the lowest error, measured as the average angular distance and that the region of low error corresponded to the range of wavelength

suitable for friction ridges. It was shown that if a previous method's orientation field [27] was replaced with that derived from the LoG filter, the positional accuracy of the minutiae in [27] was improved.

- The filtering of the orientation estimate took an alternative approach to previous methodologies. Instead of retaining unreliable estimates of the orientation and then filtering the field using and averaging technique, the approach taken section 6.3 was to remove the unreliable estimates and then later replace them with the results of a weighted neighbourhood. Although it is not the primary aim of the thesis to locate minutiae, in order to allow comparison with other techniques, it was shown that positional accuracy of the minutiae was further improved when compared with two previous techniques.
- A novel compact tree-like hierarchical representation of the palm, the nodes of which represent level sets of homogeneous variance within the orientation field, is developed in sections 7.4 for matching. The representation is unique in that it is able to represent the entire palm and that it provides an efficient means to locate regions of similar homogeneous variance to those found in the latent. It should be noted at present, only a fraction of the information contained within the tree is used in CRRA. The data structure, holding the representation, is approximately 15 times smaller than that of the palm print image.

10.1. Further Work

A number of observations were made during the thesis which could warrant further work.

- The exemplar data base contained few vestiges and only one whorl. The analysis of techniques described within the thesis would benefit from application to a much larger data base of palms and exemplars created using alternative acquisition methods. In particular, a database of forensically matched latent-exampler pairs. Such data bases are rarely used in the literature.
- It was noted in section 9.2, that latent of smaller diameter resulted in lower matching success. On closer investigation it became apparent that,
  - the current definition of a homogeneous region requires that a region be entirely enclosed within the boundary of the latent, thus many potential regions were excluded from the matching process (see red arrow in figure 9.1). This problem is exacerbated within the thenar, where the regions tend to be larger (figure 7.19).
  - individual regions matching (section 8.3), are calculated using an unweighed average and therefore, Ψ_{area}, used in *drm* and *urm* and later in seed growth, gives greater weight to larger regions. Therefore, smaller latents will have smaller regions and hence, they will be weighted less in the final matching score.

  It could therefore be advantageous in future work to include boundary regions as a special case in matching and investigate alternative weighting and normalisation methods.
- The *Hu* moments, used in seed growth, are known to be less numerically stable than other moments, such as *Chebychev* moments [96]. Given that seed growth

is a *greedy algorithm* and the results are dominated by the choice of seed, an investigation of alternative moments may be productive.

- In CRAA only a fraction of the information contained in the tree structure is used. In particular, the use of branch structure between the latent and exemplar could be used to guide a more sophisticated version of CRAA. The tree could also be extended to include representations of minutiae within the branches of the tree. This would allow the complementary use of minutiae during individual region matching.

- A *manual method* of classifying exemplar palmprints, by the frequency and position of Level I structures in the tri-radiate region, has been proposed by [18]. The method consists of the following stages:
  1. Manually identifying and classifying structure type
  2. Manually tracing the ridge flow that describes the structure
  3. Counting the number of deltas in the tri-radiate
  4. Analysing the flow of the friction ridges about loops to see if they contain deltas

  The algorithms contained in section 7.2 automatically perform the first two stages of this process. With minor modification, the remaining stages could also be performed automatically, hence making the entire method automatic.

Given the success of the further work outlined above, it would then be possible to begin to address the vexed problem of distortions in latent palm prints and their effect upon the matching process. In the hope that, the larger scale of Level I detail might alleviate the inherent difficulty of the problem.

References

[1] R. Sutton, D. Glazzard, Z. Riley, and K. Buckley, "Preliminary analysis of the nature and processing of palm marks by a u.k. fingerprint bureau." *Journal of Forensic Science*, vol. 58, no. 6, pp. 1615–1620, 2013.

[2] P. Komarinski, *Automated Fingerprint Identification Systems (AFIS)*. Elsevier Academic Press ISBN: 0-12-418351-4, 2005.

[3] M. K. Taylor, S. M. Ballou, M. F. Theofanos, and R. M. Thompson, *Latent Print Examination and Human Factors: Improving the Practice through a Systems Approach*, http://www.nist.gov/oles/upload/latent.pdf, 2012.

[4] A. Maceo, *Wiley Encyclopedia of Forensic Science*. http://dx.doi.org/10.1002/9780470061589.fsa1027: John Wiley & Sons, Ltd, 2011, ch. Friction Ridge Examination: ACE-V Documentation. [Online]. Available: http://dx.doi.org/10.1002/9780470061589.fsa1027

[5] E. H. Holder, L. O. Robinson, and J. H. Laub, *The Fingerprint Sourcebook*. http://purl.fdlp.gov/GPO/gpo18039.: U.S. Department of Justice, 2011. [Online]. Available: https://www.ncjrs.gov/pdffiles1/nij/225320.pdf

[6] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*, 2nd ed. Springer Publishing Company, Incorporated ISBN: 978-84882-253-5, 2009.

[7] D. Riley, "Palmprint identification course," senior Palmprint Expert - West Midlands Police.

[8] R. Ashbaugh, D., *Quantitative-Qualitative Friction Ridge Analysis*, Florida, Ed. CRC Press ISBN: 0-8493-7007-8, 1999.

[9] G. Langenburg and C. Champod, "Gyro system-a recommended approach to more transparent documentation," *Journal of Forensic Identification*, vol. 61, no. 4, p. 373:384, 2011. [Online]. Available: https://www.ncjrs.gov/App/Publications/abstract.aspx?ID=258112

[10] A. Moensseens, A., *Fingerprint Techniques*, N/a, Ed. Chilton Book Company ISBN: 0-8019-5527-0, 1971.

[11] M. K cken and A. C. Newell, "Fingerprint formation," *Journal of Theoretical Biology*, vol. 235, no. 1, pp. 71 – 83, 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0022519304006198

[12] H. Cummins and C. Midlo, *Finger Prints, Palms and Soles*. Dover Publications, Inc. ISBN: 978-0486207780, 1961.

[13] K. Wertheim and A. Maceo, "The critical stage of friction ridge and pattern formation," *Journal of Forensic Identification*, vol. 52 No. 1, pp. 35–85, 2002.

[14] C. Champod, C. Lennard, P. Margot, and M. Stoilovic, *Fingerprints and Other Ridge Skin Impressions*, J. Robertson, Ed. CRC Press ISBN: 0-415-27175-4, 2004.

[15] R. Ashbaugh, D., "Palmar flexion crease identification," *Journal of Forensic Identification*, vol. 41 No, 4, pp. 255–273, 1991.

[16] A. Maceo, "Qualitative assessment of skin deformation: A pilot study," *The Journal of Forensic Identification*, vol. 59 No. 4, p. 390:440, 2009.

[17] J. Dai, J. Feng, and J. Zhou, "Robust and efficient ridge-based palmprint matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 8, pp. 1618 –1632, 2012.

[18] E. Ray, "Frequency of patterns in palms," *The Journal of Forensic Identification*, vol. 6, No, 62, p. 568:587, 2012.

[19] S. K. Chatterjee, *Edgeoscopy*. Finger Print Magazine, 1962.

[20] A. Kligman, P. Zheng, and R. M. Lavker, "The anatomy and pathogenisis of wrinkles," *British Journal of Dermatology*, vol. VOL. 113 No. 1, pp. 37–42, 1985.

[21] A. Maceo, "The basis for the uniqueness and persistence of scars in the friction ridge skin," *Fingerprint Whorld*, vol. 31 No. 121, p. 147:161, 2005.

[22] L. Fang, M. K. H. Leung, T. Shikhare, V. Chan, and K. F. Choon, "Palmprint classification," in *Systems, Man and Cybernetics, 2006. SMC '06. IEEE International Conference on*, vol. 4, 2006, pp. 2965 –2969.

[23] J. Funada, N. Ohta, M. Mizoguchi, T. Temma, K. Nakanishi, A. Murai, T. Sugiuchi, T. Wakabayashi, and Y. Yamada, "Feature extraction method for palmprint considering elimination of creases," in *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, vol. 2, 1998, pp. 1849 –1854 vol.2.

[24] A. Kong, D. Zhang, and M. Kamel, "A survey of palmprint recognition," *Pattern Recognition*, vol. 42, no. 7, pp. 1408–1418, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2009.01.018

[25] D. Rutovitz, "Pattern recognition," *Journal of the Royal Statistical Society*, vol. 129, no. 4, pp. 504–530, 1966.

[26] R. Cappelli, M. Ferrara, and D. Maltoni, "Fingerprint indexing based on minutia cylinder-code," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 1051–1057, 2011.

[27] R. Cappelli, M. Ferrara, and D. Maio, "A fast and accurate palmprint recognition system based on minutiae," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 42, no. 3, pp. 956 –962, 2012.

[28] R. Wang, D. Ramos, and J. Fierrez, "Latent-to-full palmprint comparison based on radial triangulation under forensic conditions," in *Biometrics (IJCB), 2011 International Joint Conference on*, 2011, pp. 1–6.

[29] F. Turroni, D. Maltoni, R. Cappelli, and D. Maio, "Improving fingerprint orientation extraction," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 1002–1013, 2011.

[30] R. Cappelli, D. Maltoni, and F. Turroni, "Benchmarking local orientation extraction in fingerprint recognition," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, 2010, pp. 1144–1147.

[31] Y. Zheng, S. Shi, Z. Lin, Q. Wang, and Y. Zhao, "Research on offline palmprint image enhancement," in *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, vol. 1, 2007, pp. 541– 544.

[32] B. M. Mehtre, N. N. Murthy, S. Kapoor, and B. Chatterjee, "Segmentation of fingerprint images using the directional image," *Pattern Recognition*, vol. 20, no. 4, pp. 429–435, 1987. [Online]. Available: http://dx.doi.org/10.1016/0031-3203(87)90069-0

[33] Y. Zheng, Y. Liu, G. Shi, J. Li, and W. Q., "Segmentation of offline palmprint," in *Signal-Image Technologies and Internet-Based System, 2007. SITIS '07. Third International IEEE Conference on*, 2007, pp. 804–811.

[34] Z. Tan, J. Yang, Z. Shang, G. Shi, and S. Chang, "Minutiae-based offline palmprint identification system," in *Intelligent Systems, 2009. GCIS '09. WRI Global Congress on*, vol. 4, 2009, pp. 466–471.

[35] M. Vetterli, J. Kovacevic L., and K. Goyal, V., *Foundations of Signal Processing*, 978-81107038608, Ed. Cambridge University Press, 2013 (In Print).

[36] Y. Shih, F., *Image Processing and Pattern Recognition*, I. P. E. B. 978-0-470-40461-4, Ed. John Wiley & Sons Ltd, 2010, no. ISBN: 978-0-470-40461-4.

[37] T. Kamei and M. Mizoguchi, "Image filter design for fingerprint enhancement," in *Computer Vision, 1995. Proceedings., International Symposium on*, 1995, pp. 109–114.

[38] S. Chikkerur, V. Govindaraju, and N. Cartwright, A., "Fingerprint image enhancement using stft analysis," *Pattern Recognition and Image Analysis: Third International Conference on Advances in Pattern Recognition*, pp. 20–29, 2005. [Online]. Available: http://dx.doi.org/10.1007/11552499_3

[39] A. K. Jain and J. Feng, "Latent palmprint matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, No. 6, pp. 1032–1047, 2009.

[40] J. Feng, Y. Shi, and J. Zhou, "Robust and efficient algorithms for separating latent overlapped fingerprints," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 5, pp. 1498 –1510, 2012.

[41] B. Vidaydevi, G. and H. Sarojadevi, "Fingerprint ridge orientation extraction: a review of state of the art techniques," *International Journal of Computer Applications*, vol. 91, no. 3, pp. 8–13, 2014.

[42] A. M. Bazen and S. H. Gerez, "Systematic methods for the computation of the directional fields and singular points of fingerprints," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 905–919, 2002.

[43] L. Hong, Y. Wan, and J. A. K. Jain, "Fingerprint image enhancement: Algorithm and performance evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 777–789, 1998.

[44] L. O'Gorman and J. V. Nickerson, "An approach to fingerprint filter design," *Pattern Recognition*, vol. 22, no. 1, pp. 29 – 38, 1989. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0031320389900356

[45] E. Zhu, J. Yin, and G. Zhang, *Fingerprint enhancement using circular gabor filter*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 750–758. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-30126-4_91

[46] A. Pokhriyal, P. Sengar, and S. Lehri, "Fingerprint enhancement using improved orientation and circular gabor filter," *International Journal of Computer Applications*, vol. 43, no. 2, pp. 7–12, 2012.

[47] R. C. Gonzalez and R. E. Woods, *Digital Image Processing (3rd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2006, no. 978-0131687288.

[48] W. Abu-Ain, S. N. H. Sheikh, B. Bataineh, T. Abu-Ain, and K. Omar, "Skeletonization algorithm for binary images," *The 4th International Conference on Electrical Engineering and Informatics*, vol. 11, pp. 704–709, 2013.

[49] E. Liu, A. Jain, and J.Tian, "A coarse to fine minutiae-based latent palmprint matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 10, pp. 2307–2322, 2013.

[50] J. Dai and J. Zhou, "Multifeature-based high-resolution palmprint recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 945–957, 2011.

[51] M. Ahmed and R. Ward, "A rotation invariant rule-based thinning algorithm for character recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 12, pp. 1672–1678, 2002.

[52] Y. Zhang, T. and Y. Suen, C., "A fast pararllel algorithm for thinning digital patterns," *Image Processing and Computer Vision*, vol. 27 No. 3, pp. 236–240, 1984.

[53] L. Lam and C. Y. Suen, "An evaluation of parallel thinning algorithms for character recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 9, pp. 914–919, 1995.

[54] Y. Liu, Y. Zheng, G. Shi, and Q. Wang, "Research on skeletonization of palmprint image," in *Signal-Image Technologies and Internet-Based System, 2007. SITIS '07. Third International IEEE Conference on*, Dec 2007, pp. 679–686.

[55] F. Zhao and X. Tang, "Preprocessing and postprocessing for skeletopn-based fingerprint minutiae extraction," *Pattern Recognition*, vol. 40, pp. 1270–1281, 2006.

[56] A. K. Jain, L. Hong, and R. Bolle, "On-line fingerprint verification," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, no. 4, pp. 302–314, 1997.

[57] A. K. Jain, J. Feng, A. Nagar, and K. Nandakumar, "On matching latent fingerprints," in *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on*, 2008, pp. 1 –8.

[58] W. Huang, X. Lin, and X. Dai, "A novel approach for palmprint ridges features extraction," in *Image and Signal Processing, 2009. CISP '09. 2nd International Congress on*, 2009, pp. 1–5.

[59] G. Li, B. Yang, and C. Busch, "A score-level fusion fingerprint indexing approach based on minutiae vicinity and minutia cylinder-code," in *Biometrics and Forensics (IWBF), 2014 International Workshop on*, 2014, pp. 1–6.

[60] M. H. Izadi and A. Drygajlo, "Discarding low quality minutia cylinder-code pairs for improved fingerprint comparison," in *Biometrics Special Interest Group (BI-*

OSIG), 2015 International Conference of the, 2015, pp. 1–4.

[61] C. Bai, W. Wang, T. Zhao, and M. Li, "Learning compact binary quantization of minutia cylinder code," in *2016 International Conference on Biometrics (ICB)*, 2016, pp. 1–6.

[62] X. Yang, J. Feng, and J. Zhou, "Palmprint indexing based on ridge features," in *Biometrics (IJCB), 2011 International Joint Conference on*, 2011, pp. 1–8.

[63] A. K. Jain and M. Demirkus, "On latent palmprint matching," MSU, Tech. Rep., 2008.

[64] N. Duta, A. K. Jain, and K. V. Mardia, "Matching of palmprints," *Pattern Recognition Letters*, vol. 23, pp. 477–485, 2002.

[65] M. Liu and P. Yap, "Invariant representation of orientation fields for fingerprint indexing," *Pattern Recognition*, vol. 45, pp. 2532–2542, 2012.

[66] C. Huang, L. Liua, and D. Hung, "Fingerprint analysis and singular point detection," *Pattern Recognition Letters*, vol. 28, no. 15, pp. 1937–1945, 2007.

[67] M. Kawagoe and A. Tojo, "Fingerprint pattern classification," *Pattern Recognition*, vol. 17, no. 3, pp. 295–303, 1984.

[68] T. H. Le and H. T. Van, "Fingerprint reference point detection for image retrieval based on symmetry and variation," *Pattern Recognition*, vol. 45, no. 9, pp. 3360–3372, 2012.

[69] C. Hsieh, S. Shyu, and K. Hung, "An effective method for fingerprint classification," *Tamkang Journal of Science and Engineering*, vol. 12, no. 2, pp. 169–182, 2009.

[70] G. A. Bahgat, A. H. Khalil, N. S. Abdel Kadar, and S. Mashali, "Fast and accurate algorithm for core point detection in fingerprint images," *Egyptian Informatics Journal*, vol. 14, no. 1, pp. 15–25, 2013.

[71] A. Pewsey, M. Neuhauser, and G. D. Ruxton, *Circular Statistics in R*. Oxford University Press, 2013, no. 978-0-19-967113-7.

[72] K. V. Mardia and P. E. Jupp, *Directional Statistics*. John Wiley & Sons Ltd, 2000, vol. 2, no. 978-0-471-95333-3.

[73] M. Kass and A. Witkin, "Analyzing oriented patterns," *Computer Vision Graphics and Image Processing*, vol. 37, No. 3, pp. 362–3885, 1987.

[74] X. Jiang, "Extracting image orientation feature by using integration operator," *Pattern Recognition*, vol. 40, no. 2, pp. 705 – 717, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031320306001853

[75] D. Kroon, "Numerical optimization of kernel based image derivatives," 2009.

[76] H. Farid and E. P. Simoncelli, "Differentiation of discrete multidimensional signals," *IEEE Transactions on Image Processing*, vol. 13, pp. 496–508, 2004.

[77] I. Sobel. (2014) History and definition of the sobel operator.

[78] H. Scharr, *Optimal Filters for Extended Optical Flow*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 14–29. [Online]. Available: http:

//dx.doi.org/10.1007/978-3-540-69866-1_2

[79] L. Wietzke and G. Sommer, "The signal multi-vector," *J Math Imaging Vision*, vol. 37, pp. 132–150, 2010.

[80] P. Kovesi, "Phase congruency: A low-level image invariant," *Psychological Research*, vol. 64, no. 2, pp. 136–148, 2000. [Online]. Available: http://dx.doi.org/10.1007/s004260000024

[81] R. Bracewell, *Fourier Analysis and Imaging*. Springer-Scince + Business Media, LLC ISBN:978-0306481871, 2004.

[82] P. Kovesi, "Invariant measures of image features from phase information," Ph.D. dissertation, Dept. of Computer Science, University of Western Australia, 1996.

[83] L. Van Vliet, I. Young, and P. Verbeek, "Recursive gaussian derivative filters," in *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, vol. 1, 1998, pp. 509–514 vol.1.

[84] M. Felsberg and G. Sommer, "The monogenic signal," *IEEE Transactions on Signal Processing*, vol. 46, No. 12, pp. 3136–3144, 2001.

[85] O. Fleischmann, L. Wietzke, and G. Sommer, "Image analysis by conformal embedding," *Journal of Math Imaging Vision*, vol. 40, pp. 305–325, 2011.

[86] L. Moisan, "Periodic plus smooth image decomposition," *J. Math. Imaging Vis.*, vol. 39, no. 2, pp. 161–179, Feb. 2011. [Online]. Available: http://dx.doi.org/10.1007/s10851-010-0227-1

[87] D. Cremers, M. Rousson, and R. Deriche, "A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape," *International Journal of Computer Vision*, vol. 72, no. 2, pp. 195–215, 2006. [Online]. Available: http://dx.doi.org/10.1007/s11263-006-8711-1

[88] T. Kadir and M. Brady, "Unsupervised non-parametric region segmentation using level sets," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, 2003, pp. 1267–1274 vol.2.

[89] P. Soille, *Morphological Image Analysis*. Springer, 1999.

[90] S. Beucher and B. Marcotegui, "P algorithm, a dramatic enhancement of the waterfall transformation," *HAL*, 2009.

[91] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, 2011. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2010.161

[92] L. He, Y. Chao, and K. Suzuki, "A run-based two-scan labeling algorithm," *IEEE Transactions on Image Processing*, vol. 17, no. 5, pp. 749–756, 2008.

[93] M. K. Hu, "Visual pattern recognition by moment invariants," *IRE Transactions on Information Theory*, vol. 8, no. 2, pp. 179–187, 1962.

[94] H. Freeman, "On the encoding of arbitrary geometric configurations," *IRE Transactions on Electronic Computers*, vol. EC-10, no. 2, pp. 260–268, 1961.

[95] H. Bassmann and P. W. Besslich, *Ad Oculos Digital Image Processing*. Thompson Publishing, 1995.

[96] L. Kotoulas and I. Andreadis, "Fast computation of chebyshev moments," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 7, pp. 884–888, 2006.

Reference for Image

[a] Herschel, W. J. (1916). *Prints of hands and fingers made by W. J. Herschel.* [Image] Available at: http://www.sloughhistoryonline.org.uk [Accessed 17/10/2016].

Appendix A
Derivation of $\bar{\sigma}$

Laplacian of a Gaussian

$$\frac{\partial\left(\Delta f\left(x,y,\sigma\right)\right)}{\partial x} = -\frac{x\left[1-\frac{x^2+y^2}{2\sigma^2}\right]\exp\left\{-\left(\frac{x^2+y^2}{2\sigma^2}\right)\right\}}{2\pi\sigma^6} - \frac{x\exp\left\{-\left(\frac{x^2+y^2}{2\sigma^2}\right)\right\}}{2\pi\sigma^6}$$

Partial differentiating with respect to x:

$$\frac{\partial^2\left(\Delta f\left(x,y,\sigma\right)\right)}{\partial x^2} = \frac{x^2\left[1-\frac{x^2+y^2}{2\sigma^2}\right]\exp\left\{-\left(\frac{x^2}{2\sigma^2}+y^2\right)\right\}}{\pi\sigma^8} + \frac{2x^2\exp\left\{-\left(\frac{x^2}{2\sigma^2}+y^2\right)\right\}}{\pi s^8} - \frac{\left[1-\frac{x^2+y^2}{2\sigma^2}\right]\exp\left\{-\left(\frac{x^2}{2\sigma^2}+y^2\right)\right\}}{\pi\sigma^6} + \frac{2x^2\exp\left\{-\left(\frac{x^2}{2\sigma^2}+y^2\right)\right\}}{\pi\sigma^6}$$

Which simplifies to:

$$\frac{\partial^2\left(\Delta f\left(x,y,\sigma\right)\right)}{\partial x^2} = -\frac{\left(\left(x^2-\sigma^2\right)y^2+x^4-7\sigma^2x^2+4\sigma^4\right)\exp\left\{-\left(\frac{x^2}{2\sigma^2}+y^2\right)\right\}}{2\pi\sigma^{10}}$$

Which is 0, assuming $y=0$ when:

$$x^4-7\sigma^2x^2+4\sigma^4=0$$

Solving this Quartic equation for σ gives:

$$\sigma = \pm\frac{\sqrt{\left(\sqrt{33}+7\right)}x}{2^{3/2}} \pm\frac{\sqrt{\left(\sqrt{33}-7\right)}x}{2^{3/2}}$$

Which correspond to the two peaks in $\dfrac{\partial\left(\Delta f\left(x,y,\sigma\right)\right)}{\partial x}$.

Laplacian of a Poisson

Laplacian, x direction

$$\begin{aligned}\frac{\partial\left(\Delta f\left(x,y,\sigma\right)\right)}{\partial x} = & \frac{30x}{\pi\left(x^2+y^2+\sigma^2\right)^{7/2}} - \\ & \frac{52.5xy^2}{\pi\left(x^2+y^2+\sigma^2\right)^{9/2}} - \\ & \frac{52.5x^3}{\pi\left(x^2+y^2+\sigma^2\right)^{9/2}}\end{aligned}$$

Partial differentiating with respect to x:

$$\begin{aligned}\frac{\partial^2\left(\Delta f\left(x,y,\sigma\right)\right)}{\partial x^2} = & -\frac{30x}{\pi\left(x^2+y^2+\sigma^2\right)^{7/2}} + \\ & \frac{105y^2}{2\pi\left(x^2+y^2+\sigma^2\right)^{9/2}} + \\ & \frac{735x^2}{2\pi\left(x^2+y^2+\sigma^2\right)^{9/2}} - \\ & \frac{945x^2y^2}{2\pi\left(x^2+y^2+\sigma^2\right)^{11/2}} - \\ & \frac{945x^4}{2\pi\left(x^2+y^2+\sigma^2\right)^{11/2}}\end{aligned}$$

Multiplying through by $\pi\left(x^2+y^2+\sigma^2\right)^{11/2}$ gives:

$$\begin{aligned}\frac{\partial^2\left(\Delta f\left(x,y,\sigma\right)\right)}{\partial x^2} = & -30x\left(x^2+y^2+\sigma^2\right)^2 + \\ & 52.5y^2\left(x^2+y^2+\sigma^2\right) + \\ & 367.5x^2\left(x^2+y^2+\sigma^2\right) - \\ & 472.5x^2y^2 - 472.5x^4\end{aligned}$$

Assuming $y=0$, $\dfrac{\partial\left(\Delta f\left(x,y,\sigma\right)\right)}{\partial x}=0$ when:

$$-30x\left(x^2+\sigma^2\right)^2 + 367.5x^2\left(x^2+\sigma^2\right) - 472.5x^4 = 0$$

Since $\left(x^2+\sigma^2\right)^2 = x^4+2x^2\sigma^2+\sigma^4$

$$-30x^5 - 60x^3\sigma^2 - 30x\sigma^4 + 367.5x^4 + 367.5x^2\sigma^2 - 472.5x^4 = 0$$

Dividing through by x

$$-30x^4 - 60x^2\sigma^2 - 30\sigma^4 + 367.5x^3 + 367.5x\sigma^2 - 472.5x^3 = 0$$

$$-30x^4 - 60x^2\sigma^2 - 30\sigma^4 + 367.5x\sigma^2 - 105x^3 = 0$$

Solving this Quartic equation for σ gives:

$$\sigma = \begin{array}{c} \pm\dfrac{\sqrt{\left(41 - \sqrt{1393}\right)}x}{2^{3/2}} \\ \pm\dfrac{\sqrt{\left(\sqrt{1393} + 41\right)}x}{2^{3/2}} \end{array}$$

Which correspond to the two peaks in $\dfrac{\partial\left(\Delta f\left(x, y, \sigma\right)\right)}{\partial x}$